

私立文系初級 \LaTeX ユーザーが訳してみた初級 \LaTeX ユーザーのための
实用 \TeX 入門の入門〔抄〕

別冊付録

目次

第 1 章	plain T _E X	1
第 2 章	L ^A T _E X 2.09	21
	2.1 lplain.tex	23
	2.2 lhyphen.tex	42
	2.3 lfonts.tex	43
	2.4 latex.tex	56
	2.5 report.doc	175
	2.6 rep10.doc	187
第 3 章	L ^A T _E X 2 _ε	193
第 4 章	pL ^A T _E X 2 _ε	203


第1章 plain TeX

『改訂新版 TeX ブック コンピュータによる組版システム』

ドナルド・クヌース〔鷲谷好輝訳〕(アスキー・1992年)

3 TeX を扱うには (11-19 頁)

TeX は約 900 個のコントロール・シーケンスを、もともと組み込まれた命令として理解している。このコントロール・シーケンスについては、すべて本書の中で説明しているが、複雑な文書でなければ、ほとんどのコントロール・シーケンスはまったく必要ないであろう。そのため、900 個もの命令について思い悩む必要はない。さらに、実際に憶えなければならないコントロール・シーケンスは、いくつかの種類に分けることができるので、それほど無理なく理解できるだろう。数式に使われる特別な記号は記号名と同じ名前がつけられているものが多い。たとえば、 π と入力すると π が出力される。また Π は Π と入力すればよい。 \aleph は \aleph と入力し、 ∞ は ∞ と入力すればよい。また \leq は \leq 、 \neq は \neq 、 \oplus は \oplus 、 \otimes は \otimes と入力すれば出力することができる。Appendix F にこのような記号を一覧表にして載せてある。

 コントロール・シーケンスの名前では、大文字と小文字はもともと独立している。たとえば、 π と Π と Π と π は、それぞれ違うコントロール・ワードとみなされる。

TeX が理解できるのは 900 個ほどのコントロール・シーケンスだけではない。というのは、読者がさらに自分で命令を定義することが簡単にできるからである。TeX の数学記号を好きな名前に置き換えて、なじみやすくしたいときには、TeX ではそれが自由にでき、実際に置き換えてかまわない。定義の方法については、第 20 章に説明してある。

TeX がもつこうしたコントロール・シーケンスのうち、約 300 個は“プリミティブ (primitive)”と呼ばれる命令である。プリミティブは、それ以上単純な機能に

分解できない低レベルで原始的な命令である。プリミティブ以外のコントロール・シーケンスは、結局のところプリミティブをいくつか用いて定義された命令なのである。たとえば、 \input はプリミティブであるが、 \backslash や \accent はプリミティブではない。この 2 つは \backslash と \accent というプリミティブを用いて定義された命令である。

普通のユーザーが原稿の中に TeX のプリミティブを使用する機会はほとんどない。というのは、プリミティブは...そう...プリミティブだからだ。低レベルの命令を使用して TeX を扱うとすると、たくさんの命令を入力しなければならない。そうすると時間もかかるし、間違いも犯しやすくなってしまふ。そのため多くの命令を入力するよりも、どの機能を使うかを指示する高レベルなコントロール・シーケンスを使用する方がよい。こういう高レベルなコントロール・シーケンスは、プリミティブを用いて一度定義しておくだけでよいのである。たとえば、 \TeX は“TeX というロゴを組版せよ”という意味のコントロール・シーケンスであり、また \backslash は“アクセント記号を次の文字の上に置け”というコントロール・シーケンスである。確かにどちらのコントロール・シーケンスも、書体が変わればプリミティブの組合せも変える必要があるが、TeX のロゴを変える場合には、著者は、ロゴを定義している命令を 1 箇所変えるだけで済むようにしてある。そうすれば、必要なところがすべて自動的に変更されるからである。これに対して、ロゴマークが出てくるたびに、一連のプリミティブを用いて記述していたとすると、それを変更するには非常な労力を必要とするだろう。(16-17 頁)

〔中略〕

以降の章では“plain TeX”の書式について主に取り上げる。plain TeX は、Appendix B に定義されている約 600 個の基本的なコントロール・シーケンスの集まりで、300 個近くのプリミティブとともに、TeX が原稿を処理するときに利用される。TeX が約 900 個ものコントロール・シーケンスを理解して処理行くと述べたのはこのためである。TeX システムで使用できる書体を用いて、どんな要求にも合う融通性のある書式で文書を作成するときに、plain TeX がどのように使われているか

見ていくことにしよう。しかし、plain TeX は TeX のプリミティブを基にして設計された、無数にある書式の 1 つでしかないことを心に留めておいていただきたい。もっとほかの書式にしたい場合には、ユーザーが望むどんな書式でも取り扱えるように TeX を変更できるのである。だが、TeX を学ぶ最良の方法は、まず plain TeX から始めて、慣れるにしたがって少しずつ TeX の定義を変更していくことだろう。(18 頁)

Appendix B 基本的なコントロール・シーケンス (459–493 頁)

以降、この Appendix では、plain TeX の書式について詳しく説明しよう。plain TeX の書式は、通常の TeX をインプリメントしたときに、そのまま使用できるマクロの集まりである。このマクロは 3 つの基本的な目的をもっている。(1) TeX のもつプリミティブ命令で行えることは、非常に低レベルな操作でしかないため、このようなマクロを定義することによって、TeX を便利なシステムにしている。マクロが何も定義されていないバージョンの TeX システムは、現実の世界について、非常に多くのことを知らなければならない生まれたての赤ん坊のようなものである。しかし、バージョンの TeX は素早くそのようなことを憶えられるようになっている。(2) plain TeX のマクロには個人の趣味やアプリケーションに応じて、より複雑でより強力な書式に作り換えられる基本的な命令が備わっている。そのため、plain TeX を使用すると、変化に富んだ多くの文書を組めるようになっている。しかし、使い始めるとすぐに、もっと多くのことを行いたくなるであろう。(3) このマクロは、別の書式の作成方法も示す役割を果たしている。plain.tex と呼ばれるファイルが、使用しているコンピュータ・システムにあるはずである。このファイルに

は、TeX システムに何が事前に読み込まれている（ロードされている）かが定義されている。この Appendix では plain.tex ファイルの内容について述べることにしよう。しかし、ファイルのある部分はあまりにも退屈であり、また実際のマクロは、メモリ空間や実行時間を考慮して“最適化”されているので、一字一句の説明は行わないことにする。また、このマクロの最適化されていないバージョンの方が理解しやすいので、実際には最適化されたマクロではなく、最適化されていないマクロを説明することにしよう。plain.tex には、マシン上ではもっとうまく動作する同等の構成が含まれていることを忘れないでほしい。

次のようにこの Appendix B を進めていくことにしよう。plain.tex を部分ごとに少し手を加えて示しながら、詳述に値する部分にはコメントをつけ、一通り見ていくことにする。これまでに使い方を説明していないマクロ（たとえば、\vglue と \beginsection マクロは、第 1 章から第 27 章では説明していない）については、ユーザーの立場から考えてみることにしよう。しかし、ほとんどは、マクロ作成者の立場から問題を見ていくことになるだろう。(464 頁)

...として、以下のような順番で、“plain.tex”の中味について、縷々説明が行われています：

1. コード表 (L.11–)
2. レジスタの割当て (L.152–)
3. パラメータ (L.273–)
4. フォント情報 (L.393–)
5. 文書用マクロ (L.494–)
6. 数式用マクロ (L.726–)
7. 出カルーチン用マクロ (L.1140)
8. ハイフネーションとその他 (L.1218–)

説明部分を転写するのは面倒なので、以下では“plain.tex”をただ listings に流し込んでいるだけです（英語でもよければ、“texbook.tex”の 18852 行以降をそのまま読むか、または〔フォントはやってはいけないのです〕がこっそりとタイプセットして“Appendix B”を読むという手も一応あります。なお、以下の警告にもご注意ください）。

Copyright Infringement

(<http://www-cs-faculty.stanford.edu/~knuth/abcde.html>)

The source file `texbook.tex` for *The TeXbook* has been available for many years, and it begins with the following lines:

```
% This manual is copyright (C) 1984 by the American Mathematical Society.
% All rights are reserved!
% The file is distributed only for people to see its examples of TeX input,
% not for use in the preparation of books like The TeXbook.
% Permission for any other use of this file must be obtained in writing
% from the copyright holder and also from the publisher (Addison-Wesley).
\loop\iftrue
\errmessage{This manual is copyrighted and should not be TeXed}\repeat
```

From time to time, however, people have flagrantly violated these instructions, and posted PDF files of *The TeXbook* on the Internet.

Which of the words in those perfectly clear instructions do the people who do such things fail to understand? Please, if you happen to see illicit copies of these books, send a note to board@tug.org so that our user community can apply peer pressure and/or legal action to those who are unfairly exploiting our open-TeX approach.

plain.tex

```

1 % This is the plain TeX format that's described in The TeXbook.
2 % N.B.: A version number is defined at the very end of this file;
3 % please change that number whenever the file is modified!
4 % And don't modify the file unless you change its name:
5 %     Everybody's "plain.tex" file should be the same, worldwide.
6
7 % Unlimited copying and redistribution of this file are permitted as long
8 % as this file is not modified. Modifications are permitted, but only if
9 % the resulting file is not named plain.tex.
10
11 \catcode'\{=1 % left brace is begin-group character
12 \catcode'\}=2 % right brace is end-group character
13 \catcode'\$=3 % dollar sign is math shift
14 \catcode'\&=4 % ampersand is alignment tab
15 \catcode'\#=6 % hash mark is macro parameter character
16 \catcode'\^=7 \catcode'\^K=7 % circumflex and uparrow are for superscripts
17 \catcode'\_ =8 \catcode'\_A=8 % underline and downarrow are for subscripts
18 \catcode'\^I=10 % ascii tab is a blank space
19 \chardef\active=13 \catcode'\^=\active % tilde is active
20 \catcode'\^L=\active \outer\def^L{\par} % ascii form-feed is "\outer\par"
21
22 \message{Preloading the plain format: codes,}
23
24 % We had to define the \catcodes right away, before the message line,
25 % since \message uses the { and } characters.
26 % When INITEX (the TeX initializer) starts up,
27 % it has defined the following \catcode values:
28 % \catcode'\^@=9 % ascii null is ignored
29 % \catcode'\^M=5 % ascii return is end-line
30 % \catcode'\|=0 % backslash is TeX escape character
31 % \catcode'\%=14 % percent sign is comment character
32 % \catcode'\ =10 % ascii space is blank space
33 % \catcode'\^?=15 % ascii delete is invalid
34 % \catcode'\A=11 ... \catcode'\Z=11 % uppercase letters
35 % \catcode'\a=11 ... \catcode'\z=11 % lowercase letters
36 % all others are type 12 (other)
37
38 % Here is a list of the characters that have been specially catcoded:
39 \def\dospecials{\do\ \do\\\do\{\do\}\do\$\do\&%
40 \do#\do\^ \do\^K\do\_ \do\^A\do\%\do\^}
41 % (not counting ascii null, tab, linefeed, formfeed, return, delete)
42 % Each symbol in the list is preceded by \do, which can be defined
43 % if you want to do something to every item in the list.
44
45 % We make @ signs act like letters, temporarily, to avoid conflict
46 % between user names and internal control sequences of plain format.
47 \catcode'\@=11
48
49 % INITEX sets up \mathcode x=x, for x=0..255, except that
50 % \mathcode x=x+7100, for x = 'A to 'Z and 'a to 'z;
51 % \mathcode x=x+7000, for x = '0 to '9.
52 % The following changes define internal codes as recommended
53 % in Appendix C of The TeXbook:
54 \mathcode'\^@="2201 % \cdot
55 \mathcode'\^A="3223 % \downarrow
56 \mathcode'\^B="010B % \alpha
57 \mathcode'\^C="010C % \beta
58 \mathcode'\^D="225E % \land

```

```

59 \mathcode'\^^E="023A % \lnot
60 \mathcode'\^^F="3232 % \in
61 \mathcode'\^^G="0119 % \pi
62 \mathcode'\^^H="0115 % \lambda
63 \mathcode'\^^I="010D % \gamma
64 \mathcode'\^^J="010E % \delta
65 \mathcode'\^^K="3222 % \uparrow
66 \mathcode'\^^L="2206 % \pm
67 \mathcode'\^^M="2208 % \oplus
68 \mathcode'\^^N="0231 % \infty
69 \mathcode'\^^O="0140 % \partial
70 \mathcode'\^^P="321A % \subset
71 \mathcode'\^^Q="321B % \supset
72 \mathcode'\^^R="225C % \cap
73 \mathcode'\^^S="225B % \cup
74 \mathcode'\^^T="0238 % \forall
75 \mathcode'\^^U="0239 % \exists
76 \mathcode'\^^V="220A % \otimes
77 \mathcode'\^^W="3224 % \leftrightarrow
78 \mathcode'\^^X="3220 % \leftarrow
79 \mathcode'\^^Y="3221 % \rightarrow
80 \mathcode'\^^Z="8000 % \ne
81 \mathcode'\^^[="2205 % \diamond
82 \mathcode'\^^\="3214 % \le
83 \mathcode'\^^]="3215 % \ge
84 \mathcode'\^^^="3211 % \equiv
85 \mathcode'\^^_="225F % \lor
86 \mathcode'\ \_="8000 % \space
87 \mathcode'\!="5021
88 \mathcode'\ '="8000 % \prime
89 \mathcode'\(="4028
90 \mathcode'\)="5029
91 \mathcode'\*="2203 % \ast
92 \mathcode'\+="202B
93 \mathcode'\,="613B
94 \mathcode'\-="2200
95 \mathcode'\.="013A
96 \mathcode'\ /="013D
97 \mathcode'\:="303A
98 \mathcode'\;="603B
99 \mathcode'\<="313C
100 \mathcode'\=="303D
101 \mathcode'\>="313E
102 \mathcode'\?="503F
103 \mathcode'\["405B
104 \mathcode'\ \backslash="026E % \backslash
105 \mathcode'\]="505D
106 \mathcode'\_="8000 % \_
107 \mathcode'\{"4266
108 \mathcode'\|"026A
109 \mathcode'\}"5267
110 \mathcode'\^^?="1273 % \smallint
111
112 % INITEX sets \uccode'x='X and \uccode 'X='X for all letters x,
113 % and \lccode'x='x, \lccode'X='x; all other values are zero.
114 % No changes to those tables are needed in plain TeX format.
115
116 % INITEX sets \sfcode x=1000 for all x, except that \sfcode'X=999
117 % for uppercase letters. The following changes are needed:
118 \sfcode'\)=0 \sfcode'\ '=0 \sfcode'\]=0
119 % The \nonfrenchspacing macro will make further changes to \sfcode values.
120
121 % Finally, INITEX sets all \delcode values to -1, except \delcode'.=0
122 \delcode'\(="028300
123 \delcode'\)="029301
124 \delcode'\["05B302
125 \delcode'\]="05D303
126 \delcode'\<="26830A
127 \delcode'\>="26930B
128 \delcode'\ /="02F30E
129 \delcode'\|"26A30C
130 \delcode'\ \backslash="26E30F
131 % N.B. { and } should NOT get delcodes; otherwise parameter grouping fails!
132

```



```

133 % To make the plain macros more efficient in time and space,
134 % several constant values are declared here as control sequences.
135 % If they were changed, anything could happen; so they are private symbols.
136 \chardef\@ne=1
137 \chardef\tw@=2
138 \chardef\thr@@=3
139 \chardef\sixt@n=16
140 \chardef@cclv=255
141 \mathchardef\@cclvi=256
142 \mathchardef\@m=1000
143 \mathchardef\@M=10000
144 \mathchardef\@MM=20000
145
146 % Allocation of registers
147
148 % Here are macros for the automatic allocation of \count, \box, \dimen,
149 % \skip, \muskip, and \toks registers, as well as \read and \write
150 % stream numbers, \fam codes, \language codes, and \insert numbers.
151
152 \message{registers,}
153
154 % When a register is used only temporarily, it need not be allocated;
155 % grouping can be used, making the value previously in the register return
156 % after the close of the group. The main use of these macros is for
157 % registers that are defined by one macro and used by others, possibly at
158 % different nesting levels. All such registers should be defined through
159 % these macros; otherwise conflicts may occur, especially when two or more
160 % macro packages are being used at once.
161
162 % The following counters are reserved:
163 % 0 to 9 page numbering
164 % 10 count allocation
165 % 11 dimen allocation
166 % 12 skip allocation
167 % 13 muskip allocation
168 % 14 box allocation
169 % 15 toks allocation
170 % 16 read file allocation
171 % 17 write file allocation
172 % 18 math family allocation
173 % 19 language allocation
174 % 20 insert allocation
175 % 21 the most recently allocated number
176 % 22 constant -1
177 % New counters are allocated starting with 23, 24, etc. Other registers are
178 % allocated starting with 10. This leaves 0 through 9 for the user to play
179 % with safely, except that counts 0 to 9 are considered to be the page and
180 % subpage numbers (since they are displayed during output). In this scheme,
181 % \count 10 always contains the number of the highest-numbered counter that
182 % has been allocated, \count 14 the highest-numbered box, etc.
183 % Inserts are given numbers 254, 253, etc., since they require a \count,
184 % \dimen, \skip, and \box all with the same number; \count 20 contains the
185 % lowest-numbered insert that has been allocated. Of course, \box255 is
186 % reserved for \output; \count255, \dimen255, and \skip255 can be used freely.
187
188 % It is recommended that macro designers always use
189 % \global assignments with respect to registers numbered 1, 3, 5, 7, 9, and
190 % always non-\global assignments with respect to registers 0, 2, 4, 6, 8, 255.
191 % This will prevent "save stack buildup" that might otherwise occur.
192
193 \count10=22 % allocates \count registers 23, 24, ...
194 \count11=9 % allocates \dimen registers 10, 11, ...
195 \count12=9 % allocates \skip registers 10, 11, ...
196 \count13=9 % allocates \muskip registers 10, 11, ...
197 \count14=9 % allocates \box registers 10, 11, ...
198 \count15=9 % allocates \toks registers 10, 11, ...
199 \count16=-1 % allocates input streams 0, 1, ...
200 \count17=-1 % allocates output streams 0, 1, ...
201 \count18=3 % allocates math families 4, 5, ...
202 \count19=0 % allocates \language codes 1, 2, ...
203 \count20=255 % allocates insertions 254, 253, ...
204 \countdef\insc@unt=20 % the insertion counter
205 \countdef\allocationnumber=21 % the most recent allocation
206 \countdef\m@ne=22 \m@ne=-1 % a handy constant

```

```

207 \def\wlog{\immediate\write\m@ne} % write on log file (only)
208
209 % Here are abbreviations for the names of scratch registers
210 % that don't need to be allocated.
211
212 \countdef\count@=255
213 \dimendef\dimen@=0
214 \dimendef\dimen@i=1 % global only
215 \dimendef\dimen@ii=2
216 \skipdef\skip@=0
217 \toksdef\toks@=0
218
219 % Now, we define \newcount, \newbox, etc. so that you can say \newcount\foo
220 % and \foo will be defined (with \countdef) to be the next counter.
221 % To find out which counter \foo is, you can look at \allocationnumber.
222 % Since there's no \boxdef command, \chardef is used to define a \newbox,
223 % \newinsert, \newfam, and so on.
224
225 \outer\def\newcount{\alloc@0\count\countdef\insc@unt}
226 \outer\def\newdimen{\alloc@1\dimen\dimendef\insc@unt}
227 \outer\def\newskip{\alloc@2\skip\skipdef\insc@unt}
228 \outer\def\newmuskip{\alloc@3\muskip\muskipdef\@cclvi}
229 \outer\def\newbox{\alloc@4\box\chardef\insc@unt}
230 \let\newtoks=\relax % we do this to allow plain.tex to be read in twice
231 \outer\def\newhelp#1#2{\newtoks#1#1\expandafter{\csname#2\endcsname}}
232 \outer\def\newtoks{\alloc@5\toks\toksdef\@cclvi}
233 \outer\def\newread{\alloc@6\read\chardef\sixt@0n}
234 \outer\def\newwrite{\alloc@7\write\chardef\sixt@0n}
235 \outer\def\newfam{\alloc@8\fam\chardef\sixt@0n}
236 \outer\def\newlanguage{\alloc@9\language\chardef\@cclvi}
237 \def\alloc@#1#2#3#4#5{\global\advance\count1#1by\@ne
238   \ch@ck#1#4#2% make sure there's still room
239   \allocationnumber=\count1#1%
240   \global#3#5=\allocationnumber
241   \wlog{\string#5=\string#2\the\allocationnumber}}
242 \outer\def\newinsert#1{\global\advance\insc@unt by\m@ne
243   \ch@ck0\insc@unt\count
244   \ch@ck1\insc@unt\dimen
245   \ch@ck2\insc@unt\skip
246   \ch@ck4\insc@unt\box
247   \allocationnumber=\insc@unt
248   \global\chardef#1=\allocationnumber
249   \wlog{\string#1=\string\insert\the\allocationnumber}}
250 \def\ch@ck#1#2#3{\ifnum\count1#1<#2%
251   \else\errmessage{No room for a new #3}\fi}
252
253 % Here are some examples of allocation.
254 \newdimen\maxdimen \maxdimen=16383.99999pt % the largest legal <dimen>
255 \newskip\hideskip \hideskip=-1000pt plus 1fill % negative but can grow
256 \newskip\centering \centering=0pt plus 1000pt minus 1000pt
257 \newdimen\p@ \p@=1pt % this saves macro space and time
258 \newdimen\z@ \z@=0pt % can be used both for Opt and 0
259 \newskip\z@skip \z@skip=0pt plus 0pt minus 0pt
260 \newbox\voidb@x % permanently void box register
261
262 % And here's a different sort of allocation:
263 % For example, \newif\iffoo creates \footrue, \foofalse to go with \iffoo.
264 \outer\def\newif#1{\count@\escapechar \escapechar\m@ne
265   \expandafter\expandafter\expandafter
266   \def\@if#1{true}{\let#1=\iftrue}%
267   \expandafter\expandafter\expandafter
268   \def\@if#1{false}{\let#1=\iffalse}%
269   \@if#1{false}\escapechar\count@} % the condition starts out false
270 \def\@if#1#2{\csname\expandafter\if@\string#1#2\endcsname}
271 {\uccode'1='i \uccode'2='f \uppercase{\gdef\if@12{}} % 'if' is required
272
273 % Assign initial values to TeX's parameters
274
275 \message{parameters,}
276
277 % All of TeX's numeric parameters are listed here,
278 % but the code is commented out if no special value needs to be set.
279 % INITEX makes all parameters zero except where noted.
280

```

```

281 \pretolerance=100
282 \tolerance=200 % INITEX sets this to 10000
283 \hbadness=1000
284 \vbadness=1000
285 \linepenalty=10
286 \hyphenpenalty=50
287 \exhyphenpenalty=50
288 \binoppenalty=700
289 \relpenalty=500
290 \clubpenalty=150
291 \widowpenalty=150
292 \displaywidowpenalty=50
293 \brokenpenalty=100
294 \predisplaypenalty=10000
295 % \postdisplaypenalty=0
296 % \interlinepenalty=0
297 % \floatingpenalty=0, set during \insert
298 % \outputpenalty=0, set before TeX enters \output
299 \doublehyphendemerits=10000
300 \finalhyphendemerits=5000
301 \adjdemerits=10000
302 % \looseness=0, cleared by TeX after each paragraph
303 % \pausing=0
304 % \holdinginserts=0
305 % \tracingonline=0
306 % \tracingmacros=0
307 % \tracingstats=0
308 % \tracingparagraphs=0
309 % \tracingpages=0
310 % \tracingoutput=0
311 \tracinglostchars=1
312 % \tracingcommands=0
313 % \tracingrestores=0
314 % \language=0
315 \uchyph=1
316 % \lefthyphenmin=2 \righthyphenmin=3 set below
317 % \globaldefs=0
318 % \maxdeadcycles=25 % INITEX does this
319 % \hangafter=1 % INITEX does this, also TeX after each paragraph
320 % \fam=0
321 % \mag=1000 % INITEX does this
322 % \escapechar='\ % INITEX does this
323 \defaultshyphenchar='\-
324 \defaultskewchar=-1
325 % \endlinechar='\^M % INITEX does this
326 \newlinechar=-1
327 \delimitfactor=901
328 % \time=now % TeX does this at beginning of job
329 % \day=now % TeX does this at beginning of job
330 % \month=now % TeX does this at beginning of job
331 % \year=now % TeX does this at beginning of job
332 \showboxbreadth=5
333 \showboxdepth=3
334 \errorcontextlines=5
335
336 \hfuzz=0.1pt
337 \vfuzz=0.1pt
338 \overfullrule=5pt
339 \hsize=6.5in
340 \vsize=8.9in
341 \maxdepth=4pt
342 \splitmaxdepth=\maxdimen
343 \boxmaxdepth=\maxdimen
344 % \lineskiplimit=0pt, changed by \normalbaselines
345 \delimitershortfall=5pt
346 \nulldelimiterspace=1.2pt
347 \scriptspace=0.5pt
348 % \mathsurround=0pt
349 % \prelplaysize=0pt, set before TeX enters $$
350 % \displaywidth=0pt, set before TeX enters $$
351 % \displayindent=0pt, set before TeX enters $$
352 \parindent=20pt
353 % \hangindent=0pt, zeroed by TeX after each paragraph
354 % \hoffset=0pt

```

```

355 % \voffset=Opt
356
357 % \baselineskip=Opt, changed by \normalbaselines
358 % \lineskip=Opt, changed by \normalbaselines
359 \parskip=Opt plus 1pt
360 \abovedisplayskip=12pt plus 3pt minus 9pt
361 \abovedisplaysshortskip=Opt plus 3pt
362 \belowdisplayskip=12pt plus 3pt minus 9pt
363 \belowdisplaysshortskip=7pt plus 3pt minus 4pt
364 % \leftskip=Opt
365 % \rightskip=Opt
366 \topskip=10pt
367 \splittopskip=10pt
368 % \tabskip=Opt
369 % \spaceskip=Opt
370 % \xspaceskip=Opt
371 \parfillskip=Opt plus 1fil
372
373 \thinmuskip=3mu
374 \medmuskip=4mu plus 2mu minus 4mu
375 \thickmuskip=5mu plus 5mu
376
377 % We also define special registers that function like parameters:
378 \newskip\smallskipamount \smallskipamount=3pt plus 1pt minus 1pt
379 \newskip\medskipamount \medskipamount=6pt plus 2pt minus 2pt
380 \newskip\bigskipamount \bigskipamount=12pt plus 4pt minus 4pt
381 \newskip\normalbaselineskip \normalbaselineskip=12pt
382 \newskip\normallineskip \normallineskip=1pt
383 \newdimen\normallineskiplimit \normallineskiplimit=0pt
384 \newdimen\jot \jot=3pt
385 \newcount\interdisplaylinepenalty \interdisplaylinepenalty=100
386 \newcount\interfootnotelinepenalty \interfootnotelinepenalty=100
387
388 % Definitions for preloaded fonts
389
390 \def\magstephalf{1095 }
391 \def\magstep#1{\ifcase#1 \@m\or 1200\or 1440\or 1728\or 2074\or 2488\fi\relax}
392
393 % Fonts assigned to \preloaded are not part of "plain TeX",
394 % but they are preloaded so that other format packages can use them.
395 % For example, if another set of macros says "\font\ninerm=cmr9",
396 % TeX will not have to reload the font metric information for cmr9.
397
398 \message{fonts,}
399
400 \font\tenrm=cmr10 % roman text
401 \font\preloaded=cmr9
402 \font\preloaded=cmr8
403 \font\sevenrm=cmr7
404 \font\preloaded=cmr6
405 \font\fiverm=cmr5
406
407 \font\teni=cmmi10 % math italic
408 \font\preloaded=cmmi9
409 \font\preloaded=cmmi8
410 \font\seveni=cmmi7
411 \font\preloaded=cmmi6
412 \font\fivei=cmmi5
413
414 \font\tensy=cmsy10 % math symbols
415 \font\preloaded=cmsy9
416 \font\preloaded=cmsy8
417 \font\sevensy=cmsy7
418 \font\preloaded=cmsy6
419 \font\fivesy=cmsy5
420
421 \font\tenex=cmex10 % math extension
422
423 \font\preloaded=cmss10 % sans serif
424 \font\preloaded=cmssq8
425
426 \font\preloaded=cmssi10 % sans serif italic
427 \font\preloaded=cmssqi8
428

```

```

429 \font\tenbf=cmbx10 % boldface extended
430 \font\preloaded=cmbx9
431 \font\preloaded=cmbx8
432 \font\sevenbf=cmbx7
433 \font\preloaded=cmbx6
434 \font\fivebf=cmbx5
435
436 \font\tentt=cmtt10 % typewriter
437 \font\preloaded=cmtt9
438 \font\preloaded=cmtt8
439
440 \font\preloaded=cmsl10 % slanted typewriter
441
442 \font\tensl=cmsl10 % slanted roman
443 \font\preloaded=cmsl9
444 \font\preloaded=cmsl8
445
446 \font\tenit=cmti10 % text italic
447 \font\preloaded=cmti9
448 \font\preloaded=cmti8
449 \font\preloaded=cmti7
450
451 \message{more fonts,}
452 \font\preloaded=cmu10 % unslanted text italic
453
454 \font\preloaded=cmmib10 % bold math italic
455 \font\preloaded=cmbsy10 % bold math symbols
456
457 \font\preloaded=cmcsc10 % caps and small caps
458
459 \font\preloaded=cmssbx10 % sans serif bold extended
460
461 \font\preloaded=cmdunh10 % Dunhill style
462
463 \font\preloaded=cmr7 scaled \magstep4 % for titles
464 \font\preloaded=cmtt10 scaled \magstep2
465 \font\preloaded=cmssbx10 scaled \magstep2
466
467 \font\preloaded=manfnt % METAFONT logo and dragon curve and special symbols
468
469 % Additional \preloaded fonts can be specified here.
470 % (And those that were \preloaded above can be eliminated.)
471
472 \let\preloaded=\undefined % preloaded fonts must be declared anew later.
473
474 \skewchar\teni='177 \skewchar\seveni='177 \skewchar\fivei='177
475 \skewchar\teny='60 \skewchar\sevensy='60 \skewchar\fivesy='60
476
477 \textfont0=\tenrm \scriptfont0=\sevenrm \scriptscriptfont0=\fiverm
478 \def\rm{\fam\z@\tenrm}
479 \textfont1=\teni \scriptfont1=\seveni \scriptscriptfont1=\fivei
480 \def\mit{\fam\@ne} \def\oldstyle{\fam\@ne\teni}
481 \textfont2=\teny \scriptfont2=\sevensy \scriptscriptfont2=\fivesy
482 \def\cal{\fam\tw@}
483 \textfont3=\tenex \scriptfont3=\tenex \scriptscriptfont3=\tenex
484 \newfam\itfam \def\it{\fam\itfam\tenit} % it is family 4
485 \textfont\itfam=\tenit
486 \newfam\slfam \def\sl{\fam\slfam\tensl} % sl is family 5
487 \textfont\slfam=\tensl
488 \newfam\bffam \def\bf{\fam\bffam\tenbf} % bf is family 6
489 \textfont\bffam=\tenbf \scriptfont\bffam=\sevenbf
490 \scriptscriptfont\bffam=\fivebf
491 \newfam\ttfam \def\tt{\fam\ttfam\tentt} % tt is family 7
492 \textfont\ttfam=\tentt
493
494 % Macros for setting ordinary text
495 \message{macros,}
496
497 \def\frenchspacing{\sfcode'\.\@m \sfcode'\?@m \sfcode'\!@m
498 \sfcode'\:@m \sfcode'\;@m \sfcode'\,@m}
499 \def\nonfrenchspacing{\sfcode'\.3000\sfcode'\?3000\sfcode'\!3000%
500 \sfcode'\:2000\sfcode'\;1500\sfcode'\,1250 }
501
502 \def\normalbaselines{\lineskip\normallineskip

```

```

503 \baselineskip\normalbaselineskip \lineskiplimit\normallineskiplimit}
504
505 \def^^M{\ } % control <return> = control <space>
506 \def^^I{\ } % same for <tab>
507
508 \def\lq{' \def\rq{'}
509 \def\lbrack{[} \def\rbrack{]}
510
511 \let\endgraf=\par \let\endline=\cr
512
513 \def\space{ }
514 \def\empty{}
515 \def\null{\hbox{}}
516
517 \let\bgroup={ \let\egroup=}
518
519 % In \obeylines, we say '\let^^M=\par' instead of '\def^^M{\par}'
520 % since this allows, for example, '\let\par=\cr \obeylines \halign{...'
521 {\catcode^^M=\active % these lines must end with %
522 \gdef\obeylines{\catcode^^M=\active \let^^M\par}%
523 \global\let^^M\par} % this is in case ^^M appears in a \write
524 \def\obeyspaces{\catcode\ \active}
525 {\obeyspaces\global\let =\space}
526
527 \def\loop#1\repeat{\def\body{#1}\iterate}
528 \def\iterate{\body \let\next\iterate \else\let\next\relax\fi \next}
529 \let\repeat=\fi % this makes \loop... \if... \repeat skippable
530
531 \def\thinspace{\kern .16667em }
532 \def\negthinspace{\kern-.16667em }
533 \def\enspace{\kern.5em }
534
535 \def\enskip{\hskip.5em\relax}
536 \def\quad{\hskip1em\relax}
537 \def\qqquad{\hskip2em\relax}
538
539 \def\smallskip{\vskip\smallskipamount}
540 \def\medskip{\vskip\medskipamount}
541 \def\bigskip{\vskip\bigskipamount}
542
543 \def\nointerlineskip{\prevdepth-1000\p@}
544 \def\offinterlineskip{\baselineskip-1000\p@
545 \lineskipz@ \lineskiplimit\maxdimen}
546
547 \def\topglue{\nointerlineskip\vglue-\topskip\vglue} % for top of page
548 \def\vglue{\afterassignment\vgl@skip=}
549 \def\vgl@{\par \dimen@prevdepth \hrule heightz@
550 \nbreak\vskip\skip@ \prevdepth\dimen@}
551 \def\hgllue{\afterassignment\hgll@skip=}
552 \def\hgll@{\leavevmode \count@\spacefactor \vrule widthz@
553 \nbreak\hskip\skip@ \spacefactor\count@}
554
555 \def~{\penalty\@M \ } % tie
556 \def/slash/{\penalty\exhyphenpenalty} % a '/' that acts like a '-'
557
558 \def\break{\penalty-\@M}
559 \def\nobreak{\penalty \@M}
560 \def\allowbreak{\penalty z@}
561
562 \def\filbreak{\par\vfil\penalty-200\vfilneg}
563 \def\goodbreak{\par\penalty-500 }
564 \def\eject{\par\break}
565 \def\supereject{\par\penalty-\@MM}
566
567 \def\removelastskip{\ifdim\lastskip=z@\else\vskip-\lastskip\fi}
568 \def\smallbreak{\par\ifdim\lastskip<\smallskipamount
569 \removelastskip\penalty-50\smallskip\fi}
570 \def\medbreak{\par\ifdim\lastskip<\medskipamount
571 \removelastskip\penalty-100\medskip\fi}
572 \def\bigbreak{\par\ifdim\lastskip<\bigskipamount
573 \removelastskip\penalty-200\bigskip\fi}
574
575 \def\line{\hbox to\hsize}
576 \def\leftline#1{\line{#1\hss}}

```

```

577 \def\rightline#1{\line{\hss#1}}
578 \def\centerline#1{\line{\hss#1\hss}}
579
580 \def\rlap#1{\hbox to\z@{#1\hss}}
581 \def\llap#1{\hbox to\z@{\hss#1}}
582
583 \def\m@th{\mathsurround\z@}
584 \def\underbar#1{${\setbox\z@\hbox{#1}\dp\z@\z@
585 \m@th \underline{\box\z@}$}
586
587 \newbox\strutbox
588 \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}
589 \def\strut{\relax\ifmmode\copy\strutbox\else\unhcopy\strutbox\fi}
590
591 \def\hidewidth{\hskip\hideskip} % for alignment entries that can stick out
592 \def\ialign{\everycr{\}\tabskip\z@skip\halign} % initialized \halign
593 \newcount\mscount
594 \def\multispan#1{\omit \mscount#1\relax
595 \loop\ifnum\mscount>\@ne \sp@n\repeat}
596 \def\sp@n{\span\omit\advance\mscount\m@ne}
597
598 \newif\ifus@ \newif\if@cr
599 \newbox\tabs \newbox\tabsyet \newbox\tabsdone
600
601 \def\cleartabs{\global\setbox\tabsyet\@null \setbox\tabs\@null}
602 \def\settabs{\setbox\tabs\@null \futurelet\next\sett@b}
603 \let+=\relax % in case this file is being read in twice
604 \def\sett@b{\ifx\next+\def\nxt{\afterassignment\sett@b\let\nxt}%
605 \else\let\nxt\@s@tcols\fi \let\next\relax \nxt}
606 \def@s@t@b{\let\nxt\relax \us@false\m@ketabbox}
607 \def\tabalign{\us@true\m@ketabbox} % non-outer version of \+
608 \outer\def\+{\tabalign}
609 \def@s@tcols#1\columns{\count@#1\dimen@\hspace
610 \loop\ifnum\count@>\z@ \@nother \repeat}
611 \def\@nother{\dimen@ii\dimen@ \divide\dimen@ii\count@
612 \setbox\tabs\hbox{\hbox to\dimen@ii{\unhbox\tabs}}%
613 \advance\dimen@-\dimen@ii \advance\count@\m@ne}
614
615 \def\m@ketabbox{\begingroup
616 \global\setbox\tabsyet\copy\tabs
617 \global\setbox\tabsdone\@null
618 \def\cr{\@crrtrue\crr\egroup\egroup
619 \ifus@\unvbox\z@\lastbox\fi\endgroup
620 \setbox\tabs\hbox{\unhbox\tabsyet\unhbox\tabsdone}}%
621 \setbox\z@\vbox\bgroup\@crrfalse
622 \ialign\bgroup&\t@b@x##\t@b@x\crr}
623
624 \def\t@b@x{\setbox\z@\hbox\bgroup}
625 \def\t@b@x{\if@cr\egroup % now \box\z@ holds the column
626 \else\hss\egroup \global\setbox\tabsyet\hbox{\unhbox\tabsyet
627 \global\setbox\@ne\lastbox}% now \box\@ne holds its size
628 \ifvoid\@ne\global\setbox\@ne\hbox to\wd\z@}%
629 \else\setbox\z@\hbox to\wd\@ne{\unhbox\z@}\fi
630 \global\setbox\tabsdone\hbox{\box\@ne\unhbox\tabsdone}\fi
631 \box\z@}
632
633 \def\hang{\hangindent\parindent}
634 \def\textindent#1{\indent\llap{#1\enspace}\ignorespaces}
635 \def\item{\par\hang\textindent}
636 \def\itemitem{\par\indent \hangindent2\parindent \textindent}
637 \def\narrower{\advance\leftskip\parindent
638 \advance\rightskip\parindent}
639
640 \outer\def\beginsection#1\par{\vskip\z@ plus.3\vsizel\penalty-250
641 \vskip\z@ plus-.3\vsizel\bigskip\vskip\parskip
642 \message{#1}leftline{\bf#1}\nobreak\smallskip\noindent}
643 \outer\def\proclaim #1. #2\par{\medbreak
644 \noindent{\bf#1.\enspace}{\sl#2\par}}%
645 \ifdim\lastskip<\medskipamount \removelastskip\penalty55\medskip\fi}
646
647 \def\raggedright{\rightskip\z@ plus2em \spaceskip.3333em \xspaceskip.5em\relax}
648 \def\traggedright{\tt\rightskip\z@ plus2em\relax} % for use with \tt only
649
650 \chardef\%=' \%

```

```

651 \chardef \&=' \&
652 \chardef \#=' \#
653 \chardef \$=' \$
654 \chardef \ss="19
655 \chardef \ae="1A
656 \chardef \oe="1B
657 \chardef \o="1C
658 \chardef \AE="1D
659 \chardef \OE="1E
660 \chardef \O="1F
661 \chardef \i="10 \chardef \j="11 % dotless letters
662 \def \aa{\accent23a}
663 \def \l{\char32l}
664 \def \L{\leavevmode\setbox0\hbox{L}\hbox to\wd0{\hss\char32L}}
665
666 \def \leavevmode{\unhbox\voidb{x} % begins a paragraph, if necessary
667 \def \_ {\leavevmode \kern.06em \vbox{\hrule width.3em}}
668 \def \AA{\leavevmode\setbox0\hbox{!}\dimen@ \ht0\advance\dimen@-1ex%
669 \rlap{\raise.67\dimen@\hbox{\char'27}}A}
670
671 \def \mathhexbox#1#2#3{\leavevmode
672 \hbox{\m@th \mathchar"#1#2#3}}
673 \def \dag{\mathhexbox279}
674 \def \ddag{\mathhexbox27A}
675 \def \S{\mathhexbox278}
676 \def \P{\mathhexbox27B}
677 \def \Orb{\mathhexbox20D}
678
679 \def \oalign#1{\leavevmode\vtop{\baselineskip\z@skip \lineskip.25ex%
680 \ialign{##\crrc#1\crrc}} \def \o@lign{\lineskiplimit\z@ \oalign}
681 \def \ooalign{\lineskiplimit-\maxdimen \oalign} % chars over each other
682 {\catcode'p=12 \catcode't=12 \gdef\#1pt{#1}} \let\getfactor=\
683 \def \sh@ft#1{\dimen@#1\kern\expandafter\getfactor\the\fontdimen1\font
684 \dimen@} % kern by #1 times the current slant
685 \def \d#1{\o@lign{\relax#1\crrc\hidewidth\sh@ft{-1ex}.\hidewidth}}
686 \def \b#1{\o@lign{\relax#1\crrc\hidewidth\sh@ft{-3ex}%
687 \vbox to.2ex{\hbox{\char22}\vss}\hidewidth}}
688 \def \c#1{\setbox\z@\hbox{#1}\ifdim\ht\z@=1ex\accent24 #1%
689 \else\ooalign{\unhbox\z@\crrc\hidewidth\char24\hidewidth}\fi}
690 \def \copyright{\ooalign{\hfil\raise.07ex\hbox{c}\hfil\crrc\Orb}}
691
692 \def \dots{\relax\ifmmode\ldots\else\m@th\ldots\,$\fi}
693 \def \TeX{T\kern-.1667em\lower.5ex\hbox{E}\kern-.125emX}
694
695 \def \' #1{{\accent18 #1}}
696 \def \' #1{{\accent19 #1}}
697 \def \v#1{{\accent20 #1}} \let \^ \_ =\v
698 \def \u#1{{\accent21 #1}} \let \^ \^ S=\u
699 \def \# #1{{\accent22 #1}}
700 \def \~ #1{{\accent94 #1}} \let \^ \^ D=\^
701 \def \. #1{{\accent95 #1}}
702 \def \H#1{{\accent"7D #1}}
703 \def \~ #1{{\accent"7E #1}}
704 \def \# #1{{\accent"7F #1}}
705 \def \t#1{{\edef\next{\the\font}\the\textfont1\accent"7F\next#1}}
706
707 \def \hrulefill{\leaders\hrule\hfill}
708 \def \dotfill{\cleaders\hbox{\m@th \kern1.5mu.\mkern1.5mu}\hfill}
709 \def \rightarrowfill{\m@th\smash-\mkern-7mu%
710 \cleaders\hbox{\mkern-2mu\smash-\mkern-2mu}\hfill
711 \mkern-7mu\mathord\rightarrow}
712 \def \leftarrowfill{\m@th\mathord\leftarrow\mkern-7mu%
713 \cleaders\hbox{\mkern-2mu\smash-\mkern-2mu}\hfill
714 \mkern-7mu\smash-}
715 \mathchardef \bracedl="37A \mathchardef \bracerd="37B
716 \mathchardef \bracelu="37C \mathchardef \braceru="37D
717 \def \downbracefill{\m@th \setbox\z@\hbox{\bracedl}%
718 \bracedl\leaders\vrule height\ht\z@ depth\z@\hfill\braceru
719 \bracelu\leaders\vrule height\ht\z@ depth\z@\hfill\bracerd}
720 \def \upbracefill{\m@th \setbox\z@\hbox{\bracerd}%
721 \bracelu\leaders\vrule height\ht\z@ depth\z@\hfill\bracerd
722 \bracedl\leaders\vrule height\ht\z@ depth\z@\hfill\braceru}
723
724 \outer\def \bye{\par\vfill\supereject\end}

```



```

725
726 % Macros for math setting
727 \message{math definitions,}
728
729 \let\sp=^ \let\sb=_
730 \def\,\{\mskip\thinmuskip}
731 \def\>\{\mskip\medmuskip}
732 \def\;\{\mskip\thickmuskip}
733 \def\!\{\mskip-\thinmuskip}
734 \def\*{\discretionary{\thinspace\the\textfont2\char2}{-}{}}
735 {\catcode'\='=active \gdef'\bgroup\prim@s}
736 \def\prim@s{\prime\futurelet\next\pr@m@s}
737 \def\pr@m@s{\ifx'\next\let\nxt\pr@@@s \else\ifx'\next\let\nxt\pr@@@t
738 \else\let\nxt\egroup\fi\fi \nxt}
739 \def\pr@@@s#1{\prim@s} \def\pr@@@t#1#2{#2\egroup}
740 {\catcode'\^^Z=active \gdef^^Z{\not=}} % ^^Z is like \ne in math
741
742 {\catcode'\_ =active \global\let\_ =\_} % _ in math is either subscript or \_
743
744 \mathchardef\alpha="010B
745 \mathchardef\beta="010C
746 \mathchardef\gamma="010D
747 \mathchardef\delta="010E
748 \mathchardef\epsilon="010F
749 \mathchardef\zeta="0110
750 \mathchardef\eta="0111
751 \mathchardef\theta="0112
752 \mathchardef\iota="0113
753 \mathchardef\kappa="0114
754 \mathchardef\lambda="0115
755 \mathchardef\mu="0116
756 \mathchardef\nu="0117
757 \mathchardef\xi="0118
758 \mathchardef\pi="0119
759 \mathchardef\rho="011A
760 \mathchardef\sigma="011B
761 \mathchardef\tau="011C
762 \mathchardef\upsilon="011D
763 \mathchardef\phi="011E
764 \mathchardef\chi="011F
765 \mathchardef\psi="0120
766 \mathchardef\omega="0121
767 \mathchardef\varepsilon="0122
768 \mathchardef\vartheta="0123
769 \mathchardef\varpi="0124
770 \mathchardef\varrho="0125
771 \mathchardef\varsigma="0126
772 \mathchardef\varphi="0127
773 \mathchardef\Gamma="7000
774 \mathchardef\Delta="7001
775 \mathchardef\Theta="7002
776 \mathchardef\Lambda="7003
777 \mathchardef\Xi="7004
778 \mathchardef\Pi="7005
779 \mathchardef\Sigma="7006
780 \mathchardef\Upsilon="7007
781 \mathchardef\Phi="7008
782 \mathchardef\Psi="7009
783 \mathchardef\Omega="700A
784
785 \mathchardef\aleph="0240
786 \def\hbar{\mathchar'26\mkern-9\muh}}
787 \mathchardef\imath="017B
788 \mathchardef\jmath="017C
789 \mathchardef\ell="0160
790 \mathchardef\wp="017D
791 \mathchardef\Re="023C
792 \mathchardef\Im="023D
793 \mathchardef\partial="0140
794 \mathchardef\infty="0231
795 \mathchardef\prime="0230
796 \mathchardef\emptyset="023B
797 \mathchardef\nabla="0272
798 \def\surd{\mathchar"1270}}

```

```

799 \mathchardef\top="023E
800 \mathchardef\bot="023F
801 \def\angle{{\vbox{\ialign{$\m@th\scriptstyle##$\crrc
802     \not\mathrel{\mkern14mu}\crrc
803     \noalign{\nointerlineskip}
804     \mkern2.5mu\leaders\hrule height.34pt\hfill\mkern2.5mu\crrc}}}}
805 \mathchardef\triangle="0234
806 \mathchardef\forall="0238
807 \mathchardef\exists="0239
808 \mathchardef\neg="023A \let\not=\neg
809 \mathchardef\flat="015B
810 \mathchardef\natural="015C
811 \mathchardef\sharp="015D
812 \mathchardef\clubsuit="027C
813 \mathchardef\diamondsuit="027D
814 \mathchardef\heartsuit="027E
815 \mathchardef\spadesuit="027F
816
817 \mathchardef\coprod="1360
818 \mathchardef\bigvee="1357
819 \mathchardef\bigwedge="1356
820 \mathchardef\biguplus="1355
821 \mathchardef\bigcap="1354
822 \mathchardef\bigcup="1353
823 \mathchardef\intop="1352 \def\int{\intop\nolimits}
824 \mathchardef\prod="1351
825 \mathchardef\sum="1350
826 \mathchardef\bigotimes="134E
827 \mathchardef\bigoplus="134C
828 \mathchardef\bigodot="134A
829 \mathchardef\ointop="1348 \def\oint{\ointop\nolimits}
830 \mathchardef\bigsqcup="1346
831 \mathchardef\smallint="1273
832
833 \mathchardef\triangleleft="212F
834 \mathchardef\triangleright="212E
835 \mathchardef\bigtriangleup="2234
836 \mathchardef\bigtriangledown="2235
837 \mathchardef\wedge="225E \let\land=\wedge
838 \mathchardef\vee="225F \let\lor=\vee
839 \mathchardef\cap="225C
840 \mathchardef\cup="225B
841 \mathchardef\ddagger="227A
842 \mathchardef\dagger="2279
843 \mathchardef\sqcap="2275
844 \mathchardef\sqcup="2274
845 \mathchardef\uplus="225D
846 \mathchardef\amalg="2271
847 \mathchardef\diamond="2205
848 \mathchardef\bullet="220F
849 \mathchardef\wr="226F
850 \mathchardef\div="2204
851 \mathchardef\odot="220C
852 \mathchardef\oslash="220B
853 \mathchardef\otimes="220A
854 \mathchardef\ominus="2209
855 \mathchardef\oplus="2208
856 \mathchardef\mp="2207
857 \mathchardef\pm="2206
858 \mathchardef\circ="220E
859 \mathchardef\bigcirc="220D
860 \mathchardef\setminus="226E % for set difference A\setminus B
861 \mathchardef\cdot="2201
862 \mathchardef\ast="2203
863 \mathchardef\times="2202
864 \mathchardef\star="213F
865
866 \mathchardef\propto="322F
867 \mathchardef\sqsubseteq="3276
868 \mathchardef\sqsupseteq="3277
869 \mathchardef\parallel="326B
870 \mathchardef\mid="326A
871 \mathchardef\dashv="3261
872 \mathchardef\vdash="3260

```

```

873 \mathchardef\nearrow="3225
874 \mathchardef\searrow="3226
875 \mathchardef\nwarrow="322D
876 \mathchardef\swarrow="322E
877 \mathchardef\Leftrightarrow="322C
878 \mathchardef\Leftarrow="3228
879 \mathchardef\Rightarrow="3229
880 \def\neq{\not=} \let\ne=\neq
881 \mathchardef\leq="3214 \let\le=\leq
882 \mathchardef\geq="3215 \let\ge=\geq
883 \mathchardef\succ="321F
884 \mathchardef\prec="321E
885 \mathchardef\approx="3219
886 \mathchardef\succeq="3217
887 \mathchardef\preceq="3216
888 \mathchardef\supset="321B
889 \mathchardef\subset="321A
890 \mathchardef\supseteq="3213
891 \mathchardef\subseteq="3212
892 \mathchardef\in="3232
893 \mathchardef\ni="3233 \let\owns=\ni
894 \mathchardef\gg="321D
895 \mathchardef\ll="321C
896 \mathchardef\not="3236
897 \mathchardef\leftrightharpoonarrow="3224
898 \mathchardef\leftarrow="3220 \let\gets=\leftarrow
899 \mathchardef\rightarrow="3221 \let\to=\rightarrow
900 \mathchardef\mapstochar="3237 \def\mapsto{\mapstochar\rightarrow}
901 \mathchardef\sim="3218
902 \mathchardef\simseq="3227
903 \mathchardef\perp="323F
904 \mathchardef\equiv="3211
905 \mathchardef\asymp="3210
906 \mathchardef\smile="315E
907 \mathchardef\frown="315F
908 \mathchardef\leftharpoonup="3128
909 \mathchardef\leftharpoondown="3129
910 \mathchardef\rightharpoonup="312A
911 \mathchardef\rightharpoondown="312B
912
913 \def\joinrel{\mathrel{\mkern-3mu}}
914 \def\relbar{\mathrel{\smash-}} % \smash, because - has the same height as +
915 \def\Relbar{\mathrel=}
916 \mathchardef\lhook="312C \def\hookrightarrow{\lhook\joinrel\rightarrow}
917 \mathchardef\rhook="312D \def\hookleftarrow{\leftarrow\joinrel\rhook}
918 \def\bowtie{\mathrel\triangleright\joinrel\mathrel\triangleleft}
919 \def\models{\mathrel|\joinrel=}
920 \def\Longrightarrow{\Relbar\joinrel\Rightarrow}
921 \def\longrightarrow{\relbar\joinrel\rightarrow}
922 \def\longleftarrow{\leftarrow\joinrel\Relbar}
923 \def\Longleftarrow{\Leftarrow\joinrel\Relbar}
924 \def\longmapsto{\mapstochar\longrightarrow}
925 \def\longleftrightharpoonarrow{\leftarrow\joinrel\rightarrow}
926 \def\Longleftrightharpoonarrow{\Leftarrow\joinrel\Rightarrow}
927 \def\iff{\;\Longleftrightharpoonarrow\;}
928
929 \mathchardef\ldotp="613A % ldot as a punctuation mark
930 \mathchardef\cdotp="6201 % cdot as a punctuation mark
931 \mathchardef\colon="603A % colon as a punctuation mark
932 \def\ldots{\mathinner{\ldotp\ldotp\ldotp}}
933 \def\cdots{\mathinner{\cdotp\cdotp\cdotp}}
934 \def\vdots{\vbox{\baselineskip4p@ \lineskiplimitz@
935 \kern6p@\hbox{.}\hbox{.}\hbox{.}}}
936 \def\ddots{\mathinner{\mkern1mu\raise7p@\vbox{\kern7p@\hbox{.}}\mkern2mu
937 \raise4p@\hbox{.}\mkern2mu\raisep@\hbox{.}\mkern1mu}}
938
939 \def\acute{\mathaccent"7013 }
940 \def\grave{\mathaccent"7012 }
941 \def\ddot{\mathaccent"707F }
942 \def\tilde{\mathaccent"707E }
943 \def\bar{\mathaccent"7016 }
944 \def\breve{\mathaccent"7015 }
945 \def\check{\mathaccent"7014 }
946 \def\hat{\mathaccent"705E }

```

```

947 \def\vec{\mathaccent"017E }
948 \def\dot{\mathaccent"705F }
949 \def\widetilde{\mathaccent"0365 }
950 \def\widehat{\mathaccent"0362 }
951 \def\overrightarrow#1{\vbox{\m@th\ialign{##\crrc
952   \rightarrowfill\crrc\noalign{\kern-\p@\nointerlineskip}
953   $\hfil\displaystyle{#1}\hfil$\crrc}}}}
954 \def\overleftarrow#1{\vbox{\m@th\ialign{##\crrc
955   \leftarrowfill\crrc\noalign{\kern-\p@\nointerlineskip}
956   $\hfil\displaystyle{#1}\hfil$\crrc}}}}
957 \def\overbrace#1{\mathop{\vbox{\m@th\ialign{##\crrc\noalign{\kern3\p@
958   \downbracefill\crrc\noalign{\kern3\p@\nointerlineskip}
959   $\hfil\displaystyle{#1}\hfil$\crrc}}}}\limits}
960 \def\underbrace#1{\mathop{\vtop{\m@th\ialign{##\crrc
961   \upbracefill\crrc\noalign{\kern3\p@}}}}\limits}
962 \def\skew#1#2#3{\muskip\z@#1mu\divide\muskip\z@/\tw@ \mkern\muskip\z@
963   #2{\mkern-\muskip\z@#3}\mkern\muskip\z@}\mkern-\muskip\z@}}
964
965
966 \def\lmoustache{\delimiter"437A340 } % top from (, bottom from )
967 \def\rmoustache{\delimiter"537B341 } % top from ), bottom from (
968 \def\lgroup{\delimiter"462833A } % extensible ( with sharper tips
969 \def\rgroup{\delimiter"562933B } % extensible ) with sharper tips
970 \def\arrowvert{\delimiter"26A33C } % arrow without arrowheads
971 \def\Arrowvert{\delimiter"26B33D } % double arrow without arrowheads
972 \def\bracevert{\delimiter"77C33E } % the vertical bar that extends braces
973 \def\Vert{\delimiter"26B30D } \let\|=Vert
974 \def\vert{\delimiter"26A30C }
975 \def\uparrow{\delimiter"3222378 }
976 \def\downarrow{\delimiter"3223379 }
977 \def\updownarrow{\delimiter"326C33F }
978 \def\Uparrow{\delimiter"322A37E }
979 \def\Downarrow{\delimiter"322B37F }
980 \def\Updownarrow{\delimiter"326D377 }
981 \def\backslash{\delimiter"26E30F } % for double coset G\backslash H
982 \def\angle{\delimiter"526930B }
983 \def\langle{\delimiter"426830A }
984 \def\rbrace{\delimiter"5267309 } \let\}=\rbrace
985 \def\lbrace{\delimiter"4266308 } \let\{=\lbrace
986 \def\rceil{\delimiter"5265307 }
987 \def\lceil{\delimiter"4264306 }
988 \def\rfloor{\delimiter"5263305 }
989 \def\lfloor{\delimiter"4262304 }
990
991 \def\bigl{\mathopen\big}
992 \def\bigm{\mathrel\big}
993 \def\bigr{\mathclose\big}
994 \def\Bigl{\mathopen\Big}
995 \def\Bigm{\mathrel\Big}
996 \def\Bigr{\mathclose\Big}
997 \def\biggl{\mathopen\bigg}
998 \def\biggm{\mathrel\bigg}
999 \def\biggr{\mathclose\bigg}
1000 \def\Biggl{\mathopen\Bigg}
1001 \def\Biggm{\mathrel\Bigg}
1002 \def\Biggr{\mathclose\Bigg}
1003 \def\big#1{\hbox{$\left#1\vbox to8.5\p@{\right.\n@space$}}
1004 \def\Big#1{\hbox{$\left#1\vbox to11.5\p@{\right.\n@space$}}
1005 \def\bigg#1{\hbox{$\left#1\vbox to14.5\p@{\right.\n@space$}}
1006 \def\Bigg#1{\hbox{$\left#1\vbox to17.5\p@{\right.\n@space$}}
1007 \def\n@space{\null\delimiterspace\z@ \m@th}
1008
1009 \def\choose{\atopwithdelims()}
1010 \def\brack{\atopwithdelims[]}
1011 \def\brace{\atopwithdelims\{\}}
1012
1013 \def\sqrt{\radical"270370 }
1014
1015 \def\mathpalette#1#2{\mathchoice{#1\displaystyle{#2}}%
1016   {#1\textstyle{#2}}{#1\scriptstyle{#2}}{#1\scriptscriptstyle{#2}}}}
1017 \newbox\rootbox
1018 \def\root#1\of{\setbox\rootbox
1019   \hbox{$\m@th\scriptscriptstyle{#1}$}\mathpalette\root}
1020 \def\root#1#2{\setbox\z@ \hbox{$\m@th#1\sqrt{#2}$}\dimen@ \ht\z@

```

```

1021 \advance\dimen@-\dp\z@
1022 \mkern5mu\raise.6\dimen@\copy\rootbox \mkern-10mu\box\z@}
1023 \newif\ifv@ \newif\ifh@
1024 \def\vphantom{\v@true\h@false\ph@nt}
1025 \def\hphantom{\v@false\h@true\ph@nt}
1026 \def\phantom{\v@true\h@true\ph@nt}
1027 \def\ph@nt{\ifmmode\def\next{\mathpalette\mathph@nt}%
1028 \else\let\next\makeph@nt\fi\next}
1029 \def\makeph@nt#1{\setbox\z@\hbox{#1}\finph@nt}
1030 \def\mathph@nt#1#2{\setbox\z@\hbox{\$m@th#1{#2}$}\finph@nt}
1031 \def\finph@nt{\setbox\tw@null
1032 \ifv@ \ht\tw@\ht\z@ \dp\tw@\dp\z@\fi
1033 \ifh@ \wd\tw@\wd\z@\fi \box\tw@}
1034 \def\mathstrut{\vphantom{}}
1035 \def\smash\relax % \relax, in case this comes first in \halign
1036 \ifmmode\def\next{\mathpalette\mathsm@sh}\else\let\next\makesm@sh
1037 \fi\next}
1038 \def\makesm@sh#1{\setbox\z@\hbox{#1}\finsm@sh}
1039 \def\mathsm@sh#1#2{\setbox\z@\hbox{\$m@th#1{#2}$}\finsm@sh}
1040 \def\finsm@sh{\ht\z@\z@ \dp\z@\z@ \box\z@}
1041
1042 \def\cong{\mathrel{\mathpalette@vereq\sim}} % congruence sign
1043 \def@vereq#1#2{\lower.5\p@\vbox{\lineskiplimit\maxdimen\lineskip-.5\p@
1044 \ialign{\$m@th#1\hfil#\hfil$\crrc#2\crrc=\crrc}}
1045 \def\notin{\mathrel{\mathpalette@c@ncel\in}}
1046 \def@c@ncel#1#2{\m@th\oalign{\$hfil#1\mkern1mu\hfil$\crrc#1#2$}}
1047 \def\rightharpoonups{\mathrel{\mathpalette\rlh@{}}}
1048 \def\rlh@#1{\vcenter{\m@th\hbox{\oalign{\raise2pt
1049 \hbox{#1\rightharpoonup$}\crrc
1050 #1\leftharpoondown$}}}}
1051 \def\buildrel#1\over#2{\mathop{\mathop{\kern\z@#2}\limits^{#1}}}
1052 \def\doteq{\buildrel\textstyle.\over=}
1053
1054 \def\log{\mathop{\rm log}\nolimits}
1055 \def\lg{\mathop{\rm lg}\nolimits}
1056 \def\ln{\mathop{\rm ln}\nolimits}
1057 \def\lim{\mathop{\rm lim}}
1058 \def\limsup{\mathop{\rm lim}\sup}
1059 \def\liminf{\mathop{\rm lim}\inf}
1060 \def\sin{\mathop{\rm sin}\nolimits}
1061 \def\arcsin{\mathop{\rm arcsin}\nolimits}
1062 \def\sinh{\mathop{\rm sinh}\nolimits}
1063 \def\cos{\mathop{\rm cos}\nolimits}
1064 \def\arccos{\mathop{\rm arccos}\nolimits}
1065 \def\cosh{\mathop{\rm cosh}\nolimits}
1066 \def\tan{\mathop{\rm tan}\nolimits}
1067 \def\arctan{\mathop{\rm arctan}\nolimits}
1068 \def\tanh{\mathop{\rm tanh}\nolimits}
1069 \def\cot{\mathop{\rm cot}\nolimits}
1070 \def\coth{\mathop{\rm coth}\nolimits}
1071 \def\sec{\mathop{\rm sec}\nolimits}
1072 \def\csc{\mathop{\rm csc}\nolimits}
1073 \def\max{\mathop{\rm max}}
1074 \def\min{\mathop{\rm min}}
1075 \def\sup{\mathop{\rm sup}}
1076 \def\inf{\mathop{\rm inf}}
1077 \def\arg{\mathop{\rm arg}\nolimits}
1078 \def\ker{\mathop{\rm ker}\nolimits}
1079 \def\dim{\mathop{\rm dim}\nolimits}
1080 \def\hom{\mathop{\rm hom}\nolimits}
1081 \def\det{\mathop{\rm det}}
1082 \def\exp{\mathop{\rm exp}\nolimits}
1083 \def\Pr{\mathop{\rm Pr}}
1084 \def\gcd{\mathop{\rm gcd}}
1085 \def\deg{\mathop{\rm deg}\nolimits}
1086
1087 \def\bmod{\nonscript\mskip-\medmuskip\mkern5mu
1088 \mathbin{\rm mod}\penalty900\mkern5mu\nonscript\mskip-\medmuskip}
1089 \def\pmod#1{\allowbreak\mkern18mu({\rm mod}\,,\,#1)}
1090
1091 \def\cases#1{\left\{\,\vcenter{\normalbaselines\m@th
1092 \ialign{\$#\hfil$\quad#\hfil\crrc#1\crrc}}\right.}
1093 \def\matrix#1{\null\,\vcenter{\normalbaselines\m@th
1094 \ialign{\hfil$##$\hfil&\quad\hfil$##$\hfil\crrc

```

```

1095     \mathstrut\crrc\noalign{\kern-\baselineskip}
1096     #1\crrc\mathstrut\crrc\noalign{\kern-\baselineskip}}\},}
1097 \def\pmatrix#1{\left(\matrix{#1}\right)}
1098 \newdimen\p@renwd
1099 \setbox0=\hbox{\tenex B} \p@renwd=\wd0 % width of the big left (
1100 \def\bordermatrix#1{\begingroup \m@th
1101   \setbox\z@\vbox{\def\cr{\crrc\noalign{\kern2\p@global\let\cr\endline}}%
1102   \ialign{###$\hfil\kern2\p@\kern\p@renwd&\thinspace\hfil$###\hfil
1103     &&\quad\hfil$###\hfil\crrc
1104     \omit\strut\hfil\crrc\noalign{\kern-\baselineskip}}%
1105     #1\crrc\omit\strut\cr}}%
1106   \setbox\tw@\vbox{\unvcopy\z@\global\setbox\@ne\lastbox}%
1107   \setbox\tw@\hbox{\unhbox\@ne\unskip\global\setbox\@one\lastbox}%
1108   \setbox\tw@\hbox{${\kern\wd\@ne\kern-\p@renwd\left(\kern-\wd\@ne
1109     \global\setbox\@ne\vbox{\box\@ne\kern2\p@}%
1110     \vcenter{\kern-\ht\@ne\unvbox\z@\kern-\baselineskip}\,,\right)}$}%
1111   \null\;\vbox{\kern\ht\@ne\box\tw@}\endgroup}
1112
1113 \def\openup{\afterassignment\@openup\dimen@=}
1114 \def\@openup{\advance\lineskip\dimen@
1115   \advance\baselineskip\dimen@
1116   \advance\lineskiplimit\dimen@}
1117 \def\eqalign#1{\null\,,\vcenter{\openup\jot\m@th
1118   \ialign{\strut\hfil$\displaystyle{##}$&$\displaystyle{{}##}$\hfil
1119     \crrc#1\crrc}}\,,}
1120 \newif\ifdt@p
1121 \def\disply{\global\dt@ptrue\openup\jot\m@th
1122   \everycr{\noalign{\ifdt@p \global\dt@pfalse \ifdim\prevdepth>-1000\p@
1123     \vskip-\lineskiplimit \vskip\normallineskiplimit \fi
1124     \else \penalty\interdisplaylinepenalty \fi}}
1125 \def\@lign{\tabskip\z@skip\everycr{}} % restore inside \disply
1126 \def\displaylines#1{\disply \tabskip\z@skip
1127   \halign{\hbox to\displaywidth{${\@lign\hfil\displaystyle##\hfil}$}\crrc
1128     #1\crrc}}
1129 \def\eqalignno#1{\disply \tabskip\centering
1130   \halign to\displaywidth{\hfil$\@lign\displaystyle{##}$\tabskip\z@skip
1131     &${\@lign\displaystyle{{}##}$\hfil\tabskip\centering
1132     &\llap{${\@lign##}$}\tabskip\z@skip\crrc
1133     #1\crrc}}
1134 \def\leqalignno#1{\disply \tabskip\centering
1135   \halign to\displaywidth{\hfil$\@lign\displaystyle{##}$\tabskip\z@skip
1136     &${\@lign\displaystyle{{}##}$\hfil\tabskip\centering
1137     &\kern-\displaywidth\rlap{${\@lign##}$}\tabskip\displaywidth\crrc
1138     #1\crrc}}
1139
1140 % Definitions related to output
1141
1142 \message{output routines,}
1143
1144 \countdef\pageno=0 \pageno=1 % first page is number 1
1145 \newtoks\headline \headline={\hfil} % headline is normally blank
1146 \newtoks\footline \footline={\hss\tenrm\folio\hss}
1147 % footline is normally a centered page number in font \tenrm
1148 \newif\ifrggedbottom
1149 \def\raggedbottom{\topskip 10\p@ plus60\p@ \r@rggedbottomtrue}
1150 \def\normalbottom{\topskip 10\p@ \r@rggedbottomfalse} % undoes \raggedbottom
1151 \def\folio{\ifnum\pageno<\z@ \romannumeral-\pageno \else\number\pageno \fi}
1152 \def\nopagenumbers{\footline{\hfil}} % blank out the footline
1153 \def\advancepageno{\ifnum\pageno<\z@ \global\advance\pageno\m@ne
1154   \else\global\advance\pageno\@ne \fi} % increase |pageno|
1155
1156 \newinsert\footins
1157 \def\footnote#1{\let\@sf\empty % parameter #2 (the text) is read later
1158   \ifhmode\edef\@sf{\spacefactor\the\spacefactor}\/\fi
1159   #1\@sf\vfootnote{#1}}
1160 \def\vfootnote#1{\insert\footins\bgroup
1161   \interlinepenalty\interfootnotelinepenalty
1162   \splittopskip\ht\strutbox % top baseline for broken footnotes
1163   \splitmaxdepth\dp\strutbox \floatingpenalty\@MM
1164   \leftskip\z@skip \rightskip\z@skip \spaceskip\z@skip \xspaceskip\z@skip
1165   \textindent{#1}\footstrut\futurelet\next\fo@t}
1166 \def\fo@t{\ifcat\bgroup\noexpand\next \let\next\fo@t
1167   \else\let\next\fi \next}
1168 \def\fo@t{\bgroup\aftergroup\@foot\let\next}

```

```

1169 \def\ft#1{#1\@foot}
1170 \def\@foot{\strut\egroup}
1171 \def\footstrut{\vbox to\splittopskip{}}
1172 \skip\footins=\bigskipamount % space added when footnote is present
1173 \count\footins=1000 % footnote magnification factor (1 to 1)
1174 \dimen\footins=8in % maximum footnotes per page
1175
1176 \newinsert\topins
1177 \newif\ifp@ge \newif\if@mid
1178 \def\topinsert{\@midfalse\p@gefalse\@ins}
1179 \def\midinsert{\@midtrue\@ins}
1180 \def\pageinsert{\@midfalse\p@getrue\@ins}
1181 \skip\topins=\z@skip % no space added when a topinsert is present
1182 \count\topins=1000 % magnification factor (1 to 1)
1183 \dimen\topins=\maxdimen % no limit per page
1184 \def\@ins{\par\begingroup\setbox\z@\vbox\bgroup} % start a \vbox
1185 \def\endinsert{\egroup % finish the \vbox
1186   \if@mid \dimen@ht\z@ \advance\dimen@dp\z@ \advance\dimen@12\p@
1187   \advance\dimen@pagetotal \advance\dimen@-\pageshrink
1188   \ifdim\dimen@>\pagegoal\@midfalse\p@gefalse\fi\fi
1189   \if@mid \bigskip\box\z@\bigbreak
1190   \else\insert\topins{\penalty100 % floating insertion
1191     \splittopskip\z@skip
1192     \splitmaxdepth\maxdimen \floatingpenalty\z@
1193     \ifp@ge \dimen@dp\z@
1194     \vbox to\vsizel{\unvbox\z@\kern-\dimen@}% depth is zero
1195     \else \box\z@\nobreak\bigskip\fi}\fi\endgroup}
1196
1197 \output{\plainoutput}
1198 \def\plainoutput{\shipout\vbox{\makeheadline\pagebody\makefootline}}%
1199 \advancepageho
1200 \ifnum\outputpenalty>-\@MM \else\dosupereject\fi}
1201 \def\pagebody{\vbox to\vsizel{\boxmaxdepth\maxdepth \pagecontents}}
1202 \def\makeheadline{\vbox to\z@{\vskip-22.5\p@
1203   \line{\vbox to8.5\p@{\theheadline}\vss}\nointerlineskip}
1204 \def\makefootline{\baselineskip24\p@\lineskiplimit\z@line{\thefootline}}
1205 \def\dosupereject{\ifnum\insertpenalties>\z@ % something is being held over
1206   \line{\kern-\topskip\nobreak\vfill\supereject\fi}
1207
1208 \def\pagecontents{\ifvoid\topins\else\unvbox\topins\fi
1209   \dimen@=\dp\@cclv \unvbox\@cclv % open up \box255
1210   \ifvoid\footins\else % footnote info is present
1211     \vskip\skip\footins
1212     \footnoterule
1213     \unvbox\footins\fi
1214   \ifr@ggedbottom \kern-\dimen@ \vfil \fi}
1215 \def\footnoterule{\kern-3\p@
1216   \hrule width 2truein \kern 2.6\p@} % the \hrule is .4pt high
1217
1218 % Hyphenation, miscellaneous macros, and initial values for standard layout
1219 \message{hyphenation}
1220
1221 \lefthyphenmin=2 \righthyphenmin=3 % disallow x- or -xx breaks
1222 \input hyphen
1223
1224 \def\magnification{\afterassignment\m@g\count@}
1225 \def\m@g{\mag\count@}
1226 \hsize6.5truein\vsizel8.9truein\dimen\footins8truein}
1227
1228 \def\loggingall{\tracingcommands\tw@\tracingstats\tw@
1229   \tracingpages\@ne\tracingoutput\@ne\tracinglostchars\@ne
1230   \tracingmacros\tw@\tracingparagraphs\@ne\tracingrestores\@ne
1231   \showboxbreadth\maxdimen\showboxdepth\maxdimen\errorstopmode}
1232 \def\tracingall{\tracingonline\@ne\loggingall}
1233
1234 \def\showhyphens#1{\setbox0\vbox{\parfillskip\z@skip\hsize\maxdimen\tenrm
1235   \pretolerance\m@ne\tolerance\m@ne\hbadness0\showboxdepth0\ #1}}
1236
1237 \normalbaselines\rm % select roman font
1238 \nonfrenchspacing % punctuation affects the spacing
1239 \catcode'\@=12 % at signs are no longer letters
1240
1241 \def\fmtname{plain}\def\fmtversion{3.14159265} % identifies the current format

```


第2章 L^AT_EX 2.09

文書スタイルのカスタマイズ

[レスリー・ランポート(大野ほか訳)『文書処理システム L^AT_EX』(アスキー出版局・1990年)85-87頁]

L^AT_EX の標準の文書スタイルが気に入らないときには、自分の文書スタイルを作ることもできる。ここでいう“文書スタイルの変更”とは段落や箇条書などのリスト構造の出力形式を変えることで、新たな構造を作成するという意味ではない。新しい論理構造の定義方法については 3.4 節を参照のこと。

文書スタイルのカスタマイズにとりかかる前に、多くの作者が自分の文書をデザインしようとして初歩的な誤りを犯していることを肝に銘じておいてほしい。このような誤りを避けるには、熟練した組版デザインの専門家に相談するか、それに関する本を読んで勉強するほかはない。ここでは、ありがちな間違いについて注意しておく。それは、1 行が長すぎるとひどく読みづらくなるということである。この誤りを冒さないように、“句読点および空白を含めて、1 行を 75 文字以内に収める”(欧文の場合)と提案したい。

個別に文書のスタイルを変更するのなら、プリアンブルに宣言を加えるだけでよい。しかし、同じスタイル変更を複数の文書に施すのなら、新たなスタイル・オプションを作成しておいたほうが便利であろう。文書スタイル・オプションは、sty ファイルに適当な宣言を書き込めばできあがる。sty ファイルとは、ファーストネームがオプション名、拡張子が sty のファイルのことだ。たとえば、bauhaus スタイル・オプションの定義を構成する宣言を bauhaus.sty という名前のファイルに書き込んでおく。そして、\documentstyle コマンドのオプション引数に bauhaus を指定すると、L^AT_EX はメインの文書スタイルの宣言を処理した後、bauhaus.sty ファイルを読み込んでくれる。このとき、複数のスタイル・オプションが指定されていると、指定された順に対応する sty ファイルを読み込んでいく。

文書スタイル・オプションを定義した sty ファイルを読み込むときには、T_EX は @ をふつうの英文字として扱う。つまり、\@listi のように @ をコマンド名の一部として使えるのである。ユーザーが作成した文書のなかにこのようなコマンド名があると、T_EX はこれを \@記号と listi というテキスト文字列とに切り離してしまう。L^AT_EX の内部コマンドの多くは、文書中で誤って使われないように @ を含んだ名前になっている。文書スタイルのいくつかのパラメータも @ を使って定義されている。これらについての説明は、付録 C を参照のこと。

文書スタイルをすこし変えてみたいときは、ページ内のテキストの高さと横幅を決めるパラメータに手を加えるといいだろう。文書スタイルのパラメータについては、本章と付録 C で説明している。その他の変更には、L^AT_EX のコマンドの再定義が必要になる。たとえば、章番号を “Chapter 3” ではなく “Capitulo 3” と出力したいとする。そのためには、\chapter コマンドの定義を変更する必要がある。最初から \chapter コマ

ンドを定義しなければならないとしたら、T_EX と L^AT_EX 双方の内部動作に精通しているエキスパートでなければ不可能である。しかし、さいわいなことにこの場合は既存の \chapter コマンドをほんのすこし変更するだけでよい。まず、その定義がどこにあるかを調べよう。

\chapter コマンドは、文書スタイルによって定義されている。主要な文書スタイルの宣言は、スタイル・オプションと同様に sty ファイルのなかにある。たとえば、\documentstyle{report} コマンドを指定すると、T_EX は report.sty というファイルを読み込む。sty ファイルは効率優先で設計されているため、あまり読みやすいものではない。そこで、標準の L^AT_EX の sty ファイルには、それぞれに対応する doc ファイルが用意されている。このファイルは、役に立つコメントを sty ファイルに加え、読みやすいように整形したものである。つまり、report.doc ファイルは、report.sty を読みやすくした版といえる。doc ファイルがどこにあるかは、ローカルガイドを参照してほしい。

効率上の問題から、大半の L^AT_EX コマンドの定義には、3.4 節で説明した L^AT_EX のコマンドではなく、T_EX の \def コマンドを使っている。このコマンドについては、*The T_EXbook* を参照してほしい(新しい文書スタイルを最初から作成するとき以外は、\def コマンドを使用しないほうがよい。L^AT_EX のコマンドのほうが安全だし、定義の数が少なければそれらの処理にかかる余分な時間はさして気になるほどのものではない)。さて、\chapter の定義を見つけるには、まずテキストエディタを使って report.doc ファイルから “\def\chapter” という文字列を探せばいいはずだ。しかし、この文字列が見つからない…。そのファイルの冒頭部分のコメントから分かるように、report 文書スタイルは rep10.sty, rep11.sty, あるいは rep12.sty のいずれかからさらに宣言を読み込むようになっている。どのファイルを読み込むかは、文字サイズが標準の 10 ポイントか、それとも 11pt あるいは 12pt スタイル・オプションが指定されているかによって決まる。 \chapter コマンドは、これらのファイルのなかで定義されているのである。

では、10 ポイント用のスタイル・ファイルを変更するとしよう。rep10.doc ファイルのなかから “\def\chapter” という文字列を捜し、\chapter の定義を見つける。残念ながら、定義中には “Chapter” という文字列がどこで生成されているかを示すようなものは何もない。そこで、\chapter の定義に含まれているコマンドの定義、次にそれらの定義にあるコマンドの定義というふうに見つかるまで探すこともできる。しかし、“Chapter” は Chapter という入力テキストによって生成されているのだから、この 7 文字の文字列を検索するほうが簡単だ。そうすれば、\chapapp というコマンドが初期状態では Chapter と定義されており、\appendix

コマンドによって Appendix と再定義されることを示すコメントがすぐに見つかるだろう。これによって、次のようなコマンドを加えたスタイル・オプションを作成すればよいことが分かる（コマンド名に@が含まれているコマンドは、sty ファイルのなかでしか定義できない）。

```
\renewcommand{\@chapapp}{Cap\'}{\i}tulo}
```

同様に、\appendix コマンドを再定義して、Appendix を Ap\'}{e}ndice に置き換えることもできる。

この例は、コマンドを変更するときのヒントになる。環境を変更するときの手順も同様だ。ただし、ほとんどの環境は T_EX の \def コマンドで定義されている。たとえば、quote 環境は、\begin{quote}によって実行される \quote と、\end{quote}によって実行される \endquote という 2 つのコマンドとして定義されている。

L^AT_EX コマンドの定義が doc ファイルのなかにはないときは、そのコマンドはたいていは L^AT_EX の組み込みコ

マンドである。その場合、コマンドの定義は latex.tex ファイルにある。そこにもないときは、T_EX コマンドである可能性が高い。The T_EXbook を調べれば、定義を見つけられるであろう。

だが、かならずしも \chapter コマンドの例のように簡単に変更できるわけではない。コマンドによっては、T_EX コマンドについてのより高度な理解や、L^AT_EX の動作についての詳しい知識が必要になる。T_EX については、The T_EXbook を読めばすべて理解できるはずである。また、L^AT_EX について知らなければならない事柄のほとんどは、latex.tex のコメント部分に説明がある。ただし、フォント選択コマンドについては lfonts.tex に説明がある。使っているシステムではこれらのファイルがどこにあるのか、また文書スタイルのカスタマイズについての情報がほかにもあるかどうかについては、ローカルガイドを参照してほしい。

Plain T_EX コマンドを使う〔同書 210–211 頁〕

L^AT_EX は T_EX の“マクロ・パッケージ”すなわち、あらかじめ定義された T_EX コマンド群として実現されている。Plain T_EX とは、“裸”の T_EX とマクロ・パッケージ plain からなる T_EX の標準版のことである。Plain T_EX の大半のコマンドは L^AT_EX でも使えるが、利用にあたっては注意が必要である。L^AT_EX では、それ自体で 1 つのシステムとして機能するようにコマンドが統一されている。つまり、ほかの L^AT_EX コマンドと矛盾しないように使われるかぎり、そのコマンドが適正に機能するよう多くの妥協点が存在するのである。L^AT_EX のコマンドには、本書に記載されていない Plain T_EX のコマンドとともに使うと正しく動かないものもある。

Plain T_EX のコマンドが問題を起こすかどうかを知るには、実際に試してみるよりほかない。一般論としては、L^AT_EX のコマンドないし環境が使っているパラメータを変更する可能性のある Plain T_EX のコマンドは、そのコマンドや環境と組み合わせないほうがよい。たとえば、L^AT_EX のリスト作成環境のなかでは、T_EX の段落作成パラメータを変更する \hangindent のような Plain T_EX のコマンドは使ってはならない。

L^AT_EX の \output ルーチンが用いるパラメータを、本書で指定した以外の方法で変更することはできない。とくに、The T_EXbook の第 15 章に書かれていることはほとんどすべて忘れるべきであろう。とはいっても、L^AT_EX はレジスタ割当てに対する T_EX の慣習をすべて踏襲しているので、通常の T_EX コマンドを使って独自のカウンタやボックスなどを定義することは可能である。

L^AT_EX で定義が削除（ないし変更）されている Plain T_EX コマンドは下記のとおりである。ここに挙げないものは L^AT_EX コマンドで、対応する Plain T_EX 版のコマンドによく似た L^AT_EX コマンドや名前に@を含むような“内部”コマンドである。

タブ・コマンド

L^AT_EX には tabbing 環境があるので、以下のコマンドは削除した。

```
\tabs      \tabsdone   \settabs   \+
\tabset    \cleartabs  \tabalign
```

出力、脚注、図

Plain T_EX の出力ルーチンを必要とする以下のコマンドは削除されている。これらは、L^AT_EX の脚注生成コマンド、figure 環境、table 環境に置き換えられている。

```
\pageno      \nopagenumbers  \makeheadline  \footstrut
\headline    \advancepageno  \makefootline  \topins
\footline    \nopagenumbers  \dosupereject  \topinsert
\normalbottom \plainoutput     \pagecontents  \midinsert
\folio       \pagebody       \vfootnote     \pageinsert
\endinsert
```

フォント選択コマンド

L^AT_EX では、以下の Plain T_EX のコマンドは定義されていない。対応する L^AT_EX コマンドを知りたいときは、ファイル lfonts.tex を参照のこと。

```
\fivei      \fivebf      \sevensy
\fivevm     \seveni      \teni
\fivesy     \sevenbf     \oldstyle
```

方程式の位置揃え

以下の Plain T_EX コマンドは、L^AT_EX の eqnarray 環境と eqnarray*環境により不要となった。

```
\eqalign    \eqalignno    \leqalignno
```

その他

Plain T_EX の \beginsection コマンドは、L^AT_EX のセクション・コマンドに置き換えられている。また、\end コマンドと \bye コマンドは、\end{document}に置き換えられている。Plain T_EX コマンドの \centering と \line は、L^AT_EX がその名を横取りしている。Plain T_EX コマンドでできることはほとんどすべて、center 環境と flushleft 環境、flushright 環境で実現できる。Plain T_EX の \magnification コマンドにあたる L^AT_EX コマンドは存在しない。通常、出力ファイルの拡大は dvi ファイルを印字するプログラムによっておこなうことができる。

2.1 lplain.tex

```

1 % File LPLAIN - Created 29 October 1985 from plain version 1.5CM
2 %
3 % - Last modified 20 October 1988 to take into account
4 % changes to PLAIN.TEX reported by Arthur Ogawa
5 % - Modified February 8, 1990 by Dominik Wujastyk to
6 % match the PLAIN.TEX meant for TeX 3.0 (\fmtname{plain},
7 % \fmtversion{3.0}).
8 % - Modified March 15, 1990 by Frank Mittelbach to
9 % allow the use of this file both in TeX 2 and 3
10 % - Modified June 21, 1991 by RmS to clear the
11 % contents of \box0.
12 % - Modified July 1, 1991, by RmS to correct \multispan bug.
13 % - Modified August 14, 1991, By RmS to make \cases work
14 % with NFSS.
15 % - Modified October 30, 1991, by RmS to remove \catcode and
16 % \mathcode assignments for control characters.
17 % - Modified November 1, 1991, by RmS to remove ^A and ^K
18 % control characters.
19 % - Modified November 4, 1991, by RmS to add missing \m@th
20 % assignments and to introduce the file lhyphen.tex.
21 % - Modified November 7, 1991, by RmS to make it work with
22 % MLTeX version 2.
23 % - Modified March 17, 1992, by RmS to match changes in
24 % plain.tex of March 16, 1992.
25 %
26 % This is the LaTeX version of the plain TeX format that's described in
27 % The TeXbook. All modifications can be found by searching for
28 % the word 'LaTeX'.
29 % N.B.: A version number is defined at the very end of this file;
30 % please change that number whenever the file is modified!
31 % And don't modify the file under any circumstances.
32 %%
33 %% \CharacterTable
34 %% {Upper-case |A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z
35 %% Lower-case |a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
36 %% Digits |0|1|2|3|4|5|6|7|8|9
37 %% Exclamation |! Double quote |" Hash (number) |#
38 %% Dollar |$ Percent |% Ampersand |&
39 %% Acute accent |' Left paren |( Right paren |)
40 %% Asterisk |* Plus |+ Comma |,
41 %% Minus |- Point |. Solidus |/
42 %% Colon |: Semicolon |; Less than |<
43 %% Equals |= Greater than |> Question mark |?
44 %% Commercial at |@ Left bracket |[ Backslash |\  

45 %% Right bracket |] Circumflex |^ Underscore |_  

46 %% Grave accent |' Left brace |{ Vertical bar |\  

47 %% Right brace |} Tilde |~}
48 %%
49 \catcode'\{=1 % left brace is begin-group character
50 \catcode'\}=2 % right brace is end-group character
51 \catcode'\$=3 % dollar sign is math shift
52 \catcode'\&=4 % ampersand is alignment tab
53 \catcode'\#=6 % hash mark is macro parameter character
54 \catcode'\^=7 % circumflex and uparrow are for superscripts
55 \catcode'\_ =8 % underline and downarrow are for subscripts
56 \catcode'\^I=10 % ascii tab is a blank space
57 \chardef\active=13 \catcode'\^=\active % tilde is active
58 \catcode'\^L=\active \outer\def^L{\par} % ascii form-feed is "\outer\par"
59
60 \message{Preloading the plain format: codes,}
61
62 % We had to define the \catcodes right away, before the message line,
63 % since \message uses the { and } characters.
64 % When INITEX (the TeX initializer) starts up,
65 % it has defined the following \catcode values:
66 % \catcode'\^@=9 % ascii null is ignored
67 % \catcode'\^M=5 % ascii return is end-line
68 % \catcode'\|=0 % backslash is TeX escape character
69 % \catcode'\%=14 % percent sign is comment character
70 % \catcode'\ =10 % ascii space is blank space
71 % \catcode'\^?=15 % ascii delete is invalid
72 % \catcode'\A=11 ... \catcode'\Z=11 % uppercase letters

```

```

73 % \catcode'a=11 ... \catcode'z=11 % lowercase letters
74 % all others are type 12 (other)
75
76 % Here is a list of the characters that have been specially catcoded:
77 \def\dospecials{\do\ \do\\\do{\do}\do$\do\&%
78 \do#\do~\do\_ \do%\do~}
79 % (not counting ascii null, tab, linefeed, formfeed, return, delete)
80 % Each symbol in the list is preceded by \do, which can be defined
81 % if you want to do something to every item in the list.
82
83 % We make @ signs act like letters, temporarily, to avoid conflict
84 % between user names and internal control sequences of plain format.
85 \catcode@=11
86
87 % INITEX sets up \mathcode x=x, for x=0..255, except that
88 % \mathcode x=x+7100, for x = 'A to 'Z and 'a to 'z;
89 % \mathcode x=x+7000, for x = '0 to '9.
90 % The following changes define internal codes as recommended
91 % in Appendix C of The TeXbook:
92 \mathcode'\ =8000 % \space
93 \mathcode'!\ =5021
94 \mathcode'\ ' =8000 % ~\prime
95 \mathcode'\( =4028
96 \mathcode'\) =5029
97 \mathcode'\* =2203 % \ast
98 \mathcode'\+ =202B
99 \mathcode'\, =613B
100 \mathcode'\- =2200
101 \mathcode'\. =013A
102 \mathcode'\ / =013D
103 \mathcode'\: =303A
104 \mathcode'\; =603B
105 \mathcode'\< =313C
106 \mathcode'\ =303D
107 \mathcode'\> =313E
108 \mathcode'\? =503F
109 \mathcode'\[ =405B
110 \mathcode'\ \ =026E % \backslash
111 \mathcode'\ \_ =505D
112 \mathcode'\_ =8000 % \_
113 \mathcode'\{ =4266
114 \mathcode'\| =026A
115 \mathcode'\} =5267
116
117 % INITEX sets \uccode'x='X and \uccode 'X='x for all letters x,
118 % and \lccode'x='x, \lccode 'X='x; all other values are zero.
119 % No changes to those tables are needed in plain TeX format.
120
121 % INITEX sets \sfcode x=1000 for all x, except that \sfcode'X=999
122 % for uppercase letters. The following changes are needed:
123 \sfcode'\ =0 \sfcode'\_ =0 \sfcode'\ \ =0
124 % The \nonfrenchspacing macro will make further changes to \sfcode values.
125
126 % Finally, INITEX sets all \delcode values to -1, except \delcode'.=0
127 \delcode'\( =028300
128 \delcode'\) =029301
129 \delcode'\[ =05B302
130 \delcode'\] =05D303
131 \delcode'\< =26830A
132 \delcode'\> =26930B
133 \delcode'\ / =02F30E
134 \delcode'\| =26A30C
135 \delcode'\ \ =26E30F
136 % N.B. { and } should NOT get delcodes; otherwise parameter grouping fails!
137
138 % To make the plain macros more efficient in time and space,
139 % several constant values are declared here as control sequences.
140 % If they were changed, anything could happen; so they are private symbols.
141 \chardef\ne=1
142 \chardef\tw@=2
143 \chardef\thr@@=3
144 \chardef\sixt@@n=16
145 \chardef@cclv=255
146 \mathchardef@cclvi=256

```

```

147 \mathchardef\@m=1000
148 \mathchardef\@M=10000
149 \mathchardef\@MM=20000
150
151 % Allocation of registers
152
153 % Here are macros for the automatic allocation of \count, \box, \dimen,
154 % \skip, \muskip, and \toks registers, as well as \read and \write
155 % stream numbers, \fam codes, \language codes, and \insert numbers.
156
157 \message{registers,}
158
159 % When a register is used only temporarily, it need not be allocated;
160 % grouping can be used, making the value previously in the register return
161 % after the close of the group. The main use of these macros is for
162 % registers that are defined by one macro and used by others, possibly at
163 % different nesting levels. All such registers should be defined through
164 % these macros; otherwise conflicts may occur, especially when two or more
165 % more macro packages are being used at once.
166
167 % The following counters are reserved:
168 % 0 to 9 page numbering
169 % 10 count allocation
170 % 11 dimen allocation
171 % 12 skip allocation
172 % 13 muskip allocation
173 % 14 box allocation
174 % 15 toks allocation
175 % 16 read file allocation
176 % 17 write file allocation
177 % 18 math family allocation
178 % 19 language allocation
179 % 20 insert allocation
180 % 21 the most recently allocated number
181 % 22 constant -1
182 % New counters are allocated starting with 23, 24, etc. Other registers are
183 % allocated starting with 10. This leaves 0 through 9 for the user to play
184 % with safely, except that counts 0 to 9 are considered to be the page and
185 % subpage numbers (since they are displayed during output). In this scheme,
186 % \count 10 always contains the number of the highest-numbered counter that
187 % has been allocated, \count 14 the highest-numbered box, etc.
188 % Inserts are given numbers 254, 253, etc., since they require a \count,
189 % \dimen, \skip, and \box all with the same number; \count 20 contains the
190 % lowest-numbered insert that has been allocated. Of course, \box255 is
191 % reserved for \output; \count255, \dimen255, and \skip255 can be used freely.
192
193 % It is recommended that macro designers always use
194 % \global assignments with respect to registers numbered 1, 3, 5, 7, 9, and
195 % always non-\global assignments with respect to registers 0, 2, 4, 6, 8, 255.
196 % This will prevent "save stack buildup" that might otherwise occur.
197
198 \count10=22 % allocates \count registers 23, 24, ...
199 \count11=9 % allocates \dimen registers 10, 11, ...
200 \count12=9 % allocates \skip registers 10, 11, ...
201 \count13=9 % allocates \muskip registers 10, 11, ...
202 \count14=9 % allocates \box registers 10, 11, ...
203 \count15=9 % allocates \toks registers 10, 11, ...
204 \count16=-1 % allocates input streams 0, 1, ...
205 \count17=-1 % allocates output streams 0, 1, ...
206 \count18=3 % allocates math families 4, 5, ...
207 \count19=0 % allocates \language codes 1, 2, ...
208 \count20=255 % allocates insertions 254, 253, ...
209 \countdef\insc@unt=20 % the insertion counter
210 \countdef\allocationnumber=21 % the most recent allocation
211 \countdef@m@ne=22 \m@ne=-1 % a handy constant
212 \def\wlog{\immediate\write\m@ne} % write on log file (only)
213
214 % Here are abbreviations for the names of scratch registers
215 % that don't need to be allocated.
216
217 \countdef\count@=255
218 \dimendef\dimen@=0
219 \dimendef\dimen@i=1 % global only
220 \dimendef\dimen@ii=2

```

```

221 \skipdef\skip@=0
222 \toksdef\toks@=0
223
224 % Now, we define \newcount, \newbox, etc. so that you can say \newcount\foo
225 % and \foo will be defined (with \countdef) to be the next counter.
226 % To find out which counter \foo is, you can look at \allocationnumber.
227 % Since there's no \boxdef command, \chardef is used to define a \newbox,
228 % \newinsert, \newfam, and so on.
229 % LaTeX change: remove \outer from \newcount and \newdimen (FMI)
230 %       This is necessary to use \newcount inside \if...
231 %       later on.
232
233 \def\newcount{\alloc@0\count\countdef\insc@unt}
234 \def\newdimen{\alloc@1\dimen\dimendef\insc@unt}
235 \outer\def\newskip{\alloc@2\skip\skipdef\insc@unt}
236 \outer\def\newmuskip{\alloc@3\muskip\muskipdef\@cclvi}
237 \outer\def\newbox{\alloc@4\box\chardef\insc@unt}
238 \let\newtoks=\relax % we do this to allow plain.tex to be read in twice
239 \outer\def\newhelp#1#2{\newtoks#1#1\expandafter{\csname#2\endcsname}}
240 \outer\def\newtoks{\alloc@5\toks\toksdef\@cclvi}
241 \outer\def\newread{\alloc@6\read\chardef\sixt@n}
242 \outer\def\newwrite{\alloc@7\write\chardef\sixt@n}
243 \outer\def\newfam{\alloc@8\fam\chardef\sixt@n}
244 \outer\def\newlanguage{\alloc@9\language\chardef\@cclvi}
245 \def\alloc@#1#2#3#4#5{\global\advance\count1#1by\@ne
246   \ch@ck#1#4#2% make sure there's still room
247   \allocationnumber=\count1#1%
248   \global#3#5=\allocationnumber
249   \wlog{\string#5=\string#2\the\allocationnumber}}
250 \outer\def\newinsert#1{\global\advance\insc@unt by\@m@ne
251   \ch@ck0\insc@unt\count
252   \ch@ck1\insc@unt\dimen
253   \ch@ck2\insc@unt\skip
254   \ch@ck4\insc@unt\box
255   \allocationnumber=\insc@unt
256   \global\chardef#1=\allocationnumber
257   \wlog{\string#1=\string\insert\the\allocationnumber}}
258 \def\ch@ck#1#2#3{\ifnum\count1#1<#2%
259   \else\errmessage{No room for a new #3}\fi}
260
261 % Here are some examples of allocation.
262 \newdimen\maxdimen \maxdimen=16383.99999pt % the largest legal <dimen>
263 \newskip\hideskip \hideskip=-100pt plus 1fill % negative but can grow
264
265 % LaTeX change: the PLAIN \centering dimension conflicts with
266 % LaTeX's \centering command, so it is redefined to \@centering.
267 % This dimension is used by LaTeX.
268
269 \newskip\@centering \@centering=0pt plus 1000pt minus 1000pt
270
271 \newdimen\p@ \p@=1pt % this saves macro space and time
272 \newdimen\z@ \z@=0pt % can be used both for Opt and 0
273 \newskip\z@skip \z@skip=0pt plus0pt minus0pt
274 \newbox\voidb@x % permanently void box register
275
276 % And here's a different sort of allocation:
277 % For example, \newif\iffoo creates \footrue, \foofalse to go with \iffoo.
278 \outer\def\newif#1{\count@\escapechar \escapechar\@m@ne
279   \expandafter\expandafter\expandafter
280   \edef\@if#1{true}{\let\noexpand#1=noexpand\iftrue}%
281   \expandafter\expandafter\expandafter
282   \edef\@if#1{false}{\let\noexpand#1=noexpand\iffalse}%
283   \@if#1{false}\escapechar\count@} % the condition starts out false
284 \def\@if#1#2{\csname\expandafter\if@\string#1#2\endcsname}
285 {\uccode'1='i \uccode'2='f \uppercase{\gdef\if@12{}} % 'if' is required
286
287 \message{compatibility for TeX 2, }
288
289 % If this file is used in an old TeX we define the new features
290 % of TeX 3.0 as simple macros or counters so that files that uses
291 % these features can be processed in such an environment
292 % (They will however produce some other results).
293
294 \ifx\undefined\inputlineno

```

```

295 \newcount\inputlineno
296 \inputlineno-1 % this could be used to detect that
297 % an old TeX is in force
298 \ifx\undefined\language % extra test for MLTeX 2, RmS 91/11/07
299 \newcount\language
300 \fi
301 \newcount\lefthyphenmin
302 \newcount\righthyphenmin
303 \newcount\errorcontextlines
304 \newcount\holdinginserts
305 \newdimen\emergencystretch
306 \newcount\badness
307 \let\noboundary\relax
308 \newcount\setlanguage
309 \fi
310
311 % Assign initial values to TeX's parameters
312
313 \message{parameters,}
314
315 % All of TeX's numeric parameters are listed here,
316 % but the code is commented out if no special value needs to be set.
317 % INITEX makes all parameters zero except where noted.
318
319 \pretolerance=100
320 \tolerance=200 % INITEX sets this to 10000
321 \hbadness=1000
322 \vbadness=1000
323 \linepenalty=10
324 \hyphenpenalty=50
325 \exhyphenpenalty=50
326 \binoppenalty=700
327 \relpenalty=500
328 \clubpenalty=150
329 \widowpenalty=150
330 \displaywidowpenalty=50
331 \brokenpenalty=100
332 \predisplaypenalty=10000
333 % \postdisplaypenalty=0
334 % \interlinepenalty=0
335 % \floatingpenalty=0, set during \insert
336 % \outputpenalty=0, set before TeX enters \output
337 \doublehyphendemerits=10000
338 \finalhyphendemerits=5000
339 \adjdemerits=10000
340 % \looseness=0, cleared by TeX after each paragraph
341 % \pausing=0
342 % \holdinginserts=0
343 % \tracingonline=0
344 % \tracingmacros=0
345 % \tracingstats=0
346 % \tracingparagraphs=0
347 % \tracingpages=0
348 % \tracingoutput=0
349 \tracinglostchars=1
350 % \tracingcommands=0
351 % \tracingrestores=0
352 % \language=0
353 \uchyph=1
354 % \lefthyphenmin=2 \righthyphenmin=3 set below
355 % \globaldefs=0
356 % \maxdeadcycles=25 % INITEX does this
357 % \hangafter=1 % INITEX does this, also TeX after each paragraph
358 % \fam=0
359 % \mag=1000 % INITEX does this
360 % \escapechar='\ % INITEX does this
361 \defaultthyphenchar='\-
362 \defaultskewchar=-1
363 % \endlinechar='\^M % INITEX does this
364 \newlinechar=-1
365 \delimitfactor=901
366 % \time=now % TeX does this at beginning of job
367 % \day=now % TeX does this at beginning of job
368 % \month=now % TeX does this at beginning of job

```

```

369 % \year=now % TeX does this at beginning of job
370 \showboxbreadth=5
371 \showboxdepth=3
372 \errorcontextlines=5
373
374 \hfuzz=0.1pt
375 \vfuzz=0.1pt
376 \overfullrule=5pt
377 \hsize=6.5in
378 \vsize=8.9in
379 \maxdepth=4pt
380 \splitmaxdepth=\maxdimen
381 \boxmaxdepth=\maxdimen
382 % \lineskiplimit=0pt, changed by \normalbaselines
383 \delimitershortfall=5pt
384 \nulldelimiterspace=1.2pt
385 \scriptspace=0.5pt
386 % \mathsurround=0pt
387 % \predisplaysize=0pt, set before TeX enters $$
388 % \displaywidth=0pt, set before TeX enters $$
389 % \displayindent=0pt, set before TeX enters $$
390 \parindent=20pt
391 % \hangindent=0pt, zeroed by TeX after each paragraph
392 % \hoffset=0pt
393 % \voffset=0pt
394
395 % \baselineskip=0pt, changed by \normalbaselines
396 % \lineskip=0pt, changed by \normalbaselines
397 \parskip=0pt plus 1pt
398 \abovedisplayskip=12pt plus 3pt minus 9pt
399 \abovedisplayskip=0pt plus 3pt
400 \belowdisplayskip=12pt plus 3pt minus 9pt
401 \belowdisplayshortskip=7pt plus 3pt minus 4pt
402 % \leftskip=0pt
403 % \rightskip=0pt
404 \topskip=10pt
405 \splittopskip=10pt
406 % \tabskip=0pt
407 % \spaceskip=0pt
408 % \xspaceskip=0pt
409 \parfillskip=0pt plus 1fil
410
411 \thinmuskip=3mu
412 \medmuskip=4mu plus 2mu minus 4mu
413 \thickmuskip=5mu plus 5mu
414
415 % We also define special registers that function like parameters:
416 \newskip\smallskipamount \smallskipamount=3pt plus 1pt minus 1pt
417 \newskip\medskipamount \medskipamount=6pt plus 2pt minus 2pt
418 \newskip\bigskipamount \bigskipamount=12pt plus 4pt minus 4pt
419 \newskip\normalbaselineskip \normalbaselineskip=12pt
420 \newskip\normallineskip \normallineskip=1pt
421 \newdimen\normallineskiplimit \normallineskiplimit=0pt
422 \newdimen\jot \jot=3pt
423 \newcount\interdisplaylinepenalty \interdisplaylinepenalty=100
424 \newcount\interfootnotelinepenalty \interfootnotelinepenalty=100
425
426 % Definitions for preloaded fonts
427
428 \def\magstephalf{1095 }
429 \def\magstep#1{\ifcase#1 \@m\or 1200\or 1440\or 1728\or 2074\or 2488\fi\relax}
430
431 % Fonts assigned to \preloaded are not part of "plain TeX",
432 % but they are preloaded so that other format packages can use them.
433 % For example, if another set of macros says "\font\ninerm=cmr9",
434 % TeX will not have to reload the font metric information for cmr9.
435
436 % LaTeX font definitions are taken from the file LFONTS.TEX,
437 % so all of PLAIN's font definitions are commented out.
438
439 %\message{fonts,}
440 %
441 %\font\tenrm=cmr10 % roman text
442 %\font\preloaded=cmr9

```



```

443 %\font\preloaded=cmr8
444 %\font\sevenrm=cmr7
445 %\font\preloaded=cmr6
446 %\font\fivevm=cmr5
447 %
448 %\font\teni=cmmi10 % math italic
449 %\font\preloaded=cmmi9
450 %\font\preloaded=cmmi8
451 %\font\seveni=cmmi7
452 %\font\preloaded=cmmi6
453 %\font\fivei=cmmi5
454 %
455 %\font\tensy=cmsy10 % math symbols
456 %\font\preloaded=cmsy9
457 %\font\preloaded=cmsy8
458 %\font\sevensy=cmsy7
459 %\font\preloaded=cmsy6
460 %\font\fivesy=cmsy5
461 %
462 %\font\tenex=cmex10 % math extension
463 %
464 %\font\preloaded=cmss10 % sans serif
465 %\font\preloaded=cmssq8
466 %
467 %\font\preloaded=cmssi10 % sans serif italic
468 %\font\preloaded=cmssqi8
469 %
470 %\font\tenbf=cmbx10 % boldface extended
471 %\font\preloaded=cmbx9
472 %\font\preloaded=cmbx8
473 %\font\sevenbf=cmbx7
474 %\font\preloaded=cmbx6
475 %\font\fivebf=cmbx5
476 %
477 %\font\tentt=cmtt10 % typewriter
478 %\font\preloaded=cmtt9
479 %\font\preloaded=cmtt8
480 %
481 %\font\preloaded=cmslitt10 % slanted typewriter
482 %
483 %\font\tensl=cmsl10 % slanted roman
484 %\font\preloaded=cmsl9
485 %\font\preloaded=cmsl8
486 %
487 %\font\tenit=cmti10 % text italic
488 %\font\preloaded=cmti9
489 %\font\preloaded=cmti8
490 %\font\preloaded=cmti7
491 %
492 %\message{more fonts,}
493 %\font\preloaded=cmu10 % unslanted text italic
494 %
495 %\font\preloaded=cmmib10 % bold math italic
496 %\font\preloaded=cmbsy10 % bold math symbols
497 %
498 %\font\preloaded=cmcsc10 % caps and small caps
499 %
500 %\font\preloaded=cmssbx10 % sans serif bold extended
501 %
502 %\font\preloaded=cmdunh10 % Dunhill style
503 %
504 %\font\preloaded=cmr7 scaled \magstep4 % for titles
505 %\font\preloaded=cmtt10 scaled \magstep2
506 %\font\preloaded=cmssbx10 scaled \magstep2
507 %
508 %\font\preloaded=manfnt % METAFONT logo and dragon curve and special symbols
509 %
510 %% Additional \preloaded fonts can be specified here.
511 %% (And those that were \preloaded above can be eliminated.)
512 %
513 %\let\preloaded=\undefined % preloaded fonts must be declared anew later.
514 %
515 %\skewchar\teni='177 \skewchar\seveni='177 \skewchar\fivei='177
516 %\skewchar\tensy='60 \skewchar\sevensy='60 \skewchar\fivesy='60

```

```

517 %
518 %\textfont0=\tenrm \scriptfont0=\sevenrm \scriptscriptfont0=\fiverm
519 %\def\rm{\fam\z@\tenrm}
520 %\textfont1=\teni \scriptfont1=\seveni \scriptscriptfont1=\fivei
521 %\def\mit{\fam@ne} \def\oldstyle{\fam@ne\teni}
522 %\textfont2=\tensty \scriptfont2=\sevensty \scriptscriptfont2=\fivesy
523 %\def\cal{\fam\tw@}
524 %\textfont3=\tenex \scriptfont3=\tenex \scriptscriptfont3=\tenex
525 %\newfam\itfam \def\it{\fam\itfam\tenit} % \it is family 4
526 %\textfont\itfam=\tenit
527 %\newfam\slfam \def\sl{\fam\slfam\tensl} % \sl is family 5
528 %\textfont\slfam=\tensl
529 %\newfam\bfam \def\bf{\fam\bfam\tenbf} % \bf is family 6
530 %\textfont\bfam=\tenbf \scriptfont\bfam=\sevenbf
531 %\scriptscriptfont\bfam=\fivebf
532 %\newfam\ttfam \def\tt{\fam\ttfam\tentt} % \tt is family 7
533 %\textfont\ttfam=\tentt
534
535 % Macros for setting ordinary text
536 \message{macros,}
537
538 \def\frenchspacing{\sfcode'\.\@m \sfcode'\?@m \sfcode'\!@m
539 \sfcode'\:@m \sfcode'\;@m \sfcode'\,@m}
540 \def\nonfrenchspacing{\sfcode'\.3000\sfcodes'\?3000\sfcodes'\!3000%
541 \sfcode'\:2000\sfcodes'\;1500\sfcodes'\,@1250 }
542
543 \def\normalbaselines{\lineskip\normallineskip
544 \baselineskip\normalbaselineskip \lineskiplimit\normallineskiplimit}
545
546 \def^^M{\ } % control <return> = control <space>
547 \def^^I{\ } % same for <tab>
548
549 \def\lq{'} \def\rq{'}
550 \def\lbrack{[} \def\rbrack{]}
551
552 \let\endgraf=\par \let\endline=\cr
553
554 \def\space{ }
555 \def\empty{}
556 \def\null{\hbox{}}
557
558 \let\bgroup={ \let\egroup=}
559
560 % In \obeylines, we say '\let^^M=\par' instead of '\def^^M{\par}'
561 % since this allows, for example, '\let\par=\cr \obeylines \halign{...}'
562 {\catcode'\^^M=\active % these lines must end with %
563 \gdef\obeylines{\catcode'\^^M\active \let^^M\par}%
564 \global\let^^M\par} % this is in case ^^M appears in a \write
565 \def\obeyspaces{\catcode'\ \active}
566 {\obeyspaces\global\let =\space}
567
568 \def\loop#1\repeat{\def\body{#1}\iterate}
569 \def\iterate{\body \let\next\iterate \else\let\next\relax\fi \next}
570 \let\repeat=\fi % this makes \loop... \if... \repeat skippable
571
572 \def\thinspace{\kern .16667em }
573 \def\negthinspace{\kern-.16667em }
574 \def\enspace{\kern.5em }
575
576 \def\enskip{\hskip.5em\relax}
577 \def\quad{\hskip1em\relax}
578 \def\qqquad{\hskip2em\relax}
579
580 \def\smallskip{\vskip\smallskipamount}
581 \def\medskip{\vskip\medskipamount}
582 \def\bigskip{\vskip\bigskipamount}
583
584 \def\nointerlineskip{\prevdepth-1000\p@}
585 \def\offinterlineskip{\baselineskip-1000\p@
586 \lineskip\z@ \lineskiplimit\maxdimen}
587
588 \def\vglue{\afterassignment\vgl@\skip@=}
589 \def\vgl@{\par \dimen@prevdepth \hrule height\z@
590 \nobreak\vskip\skip@ \prevdepth\dimen@}

```

```

591 \def\hglue{\afterassignment\hgl@skip@=}
592 \def\hgl@{\leavevmode \count@spacefactor \vrule width\z@
593   \nobreak\hskip\skip@ \spacefactor\count@}
594
595 \def~{\penalty\@M \ } % tie
596 \def\slash{/\penalty\exhyphenpenalty} % a '/' that acts like a '-'
597
598 \def\break{\penalty-\@M}
599 \def\nobreak{\penalty \@M}
600 \def\allowbreak{\penalty \z@}
601
602 \def\filbreak{\par\vfil\penalty-200\vfilneg}
603 \def\goodbreak{\par\penalty-500 }
604 \def\eject{\par\break}
605 \def\supereject{\par\penalty-\@MM}
606
607 \def\removelastskip{\ifdim\lastskip=\z@else\vskip-\lastskip\fi}
608 \def\smallbreak{\par\ifdim\lastskip<\smallskipamount
609   \removelastskip\penalty-50\smallskip\fi}
610 \def\medbreak{\par\ifdim\lastskip<\medskipamount
611   \removelastskip\penalty-100\medskip\fi}
612 \def\bigbreak{\par\ifdim\lastskip<\bigskipamount
613   \removelastskip\penalty-200\bigskip\fi}
614
615 % \line changed to @@line because LaTeX redefines \line
616 \def@@line{\hbox to\hsize}
617 \def\leftline#1{\@@line{#1\hss}}
618 \def\rightline#1{\@@line{\hss#1}}
619 \def\centerline#1{\@@line{\hss#1\hss}}
620 % end of LaTeX \line -> @@line change
621
622 \def\rlap#1{\hbox to\z@{#1\hss}}
623 \def\llap#1{\hbox to\z@{\hss#1}}
624
625 \def\m@th{\mathsurround=\z@}
626 \def\underbar#1{\$setbox\z@\hbox{#1}\dp\z@\z@
627   \m@th \underline{\box\z@}\$}
628
629 \newbox\strutbox
630 \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}
631 \def\strut{\relax\ifmmode\copy\strutbox\else\unhcopy\strutbox\fi}
632
633 \def\hidewidth{\hskip\hideskip} % for alignment entries that can stick out
634 \def\ialign{\everycr{\}\tabskip\z@skip\halign} % initialized \halign
635 \newcount\mscount
636 \def\multispan#1{\omit \mscount#1\relax
637   \loop\ifnum\mscount>\@ne \sp@n\repeat}
638 \def\sp@n{\span\omit\advance\mscount\@one}
639
640 % LaTeX has its own tabbing environment, so PLAIN's is disabled.
641 %
642 %\newif\ifus@ \newif\if@cr
643 %\newbox\tabs \newbox\tabsyet \newbox\tabsdone
644 %
645 %\def\cleartabs{\global\setbox\tabsyet\null \setbox\tabs\null}
646 %\def\settabs{\setbox\tabs\null \futurelet\next\sett@b}
647 %\let\+=\relax % in case this file is being read in twice
648 %\def\sett@b{\ifx\next+\let\next\relax
649 %   \def\next{\afterassignment\s@tt@b\let\next}%
650 %   \else\let\next\s@tcols\fi\next}
651 %\def\s@tt@b{\let\next\relax\us@false\m@ketabbox}
652 %\def\tabalign{\us@true\m@ketabbox} % non-outer version of \+
653 %\outer\def\+{\tabalign}
654 %\def\s@tcols#1\columns{\count@#1 \dimen@\hsize
655 %   \loop\ifnum\count@>\z@ \@nother \repeat}
656 %\def\@nother{\dimen@ii\dimen@ \divide\dimen@ii\count@
657 %   \setbox\tabs\hbox{\hbox to\dimen@ii}\unhbox\tabs}%
658 %   \advance\dimen@-\dimen@ii \advance\count@\@one}
659 %
660 %\def\m@ketabbox{\begingroup
661 %   \global\setbox\tabsyet\copy\tabs
662 %   \global\setbox\tabsdone\null
663 %   \def\cr{\@crtrue\cr\egroup\egroup
664 %     \ifus@\unvbox\z@\lastbox\fi\endgroup

```

```

665 % \setbox\tabs\hbox{\unhbox\tabsyet\unhbox\tabsdone}}%
666 % \setbox\z@\vbox\bgroup\@crfalse
667 % \ialign\bgroup&\t@bbbox##\t@bb@x\crrc}
668 %
669 %\def\t@bbbox{\setbox\z@\hbox\bgroup}
670 %\def\t@bb@x{\if@cr\egroup % now \box\z@ holds the column
671 % \else\hss\egroup \global\setbox\tabsyet\hbox{\unhbox\tabsyet
672 % \global\setbox\@ne\lastbox}% now \box\@ne holds its size
673 % \ifvoid\@ne\global\setbox\@ne\hbox to\wd\z@{}}%
674 % \else\setbox\z@\hbox to\wd\@ne{\unhbox\z@}\fi
675 % \global\setbox\tabsdone\hbox{\box\@ne\unhbox\tabsdone}\fi
676 % \box\z@}
677
678 \def\hang{\hangindent\parindent}
679 \def\textindent#1{\indent\llap{#1\enspace}\ignorespaces}
680 \def\item{\par\hang\textindent}
681 % RmS 91/11/04: Removed \itemitem since never needed/useful on LaTeX.
682 %\def\itemitem{\par\indent \hangindent2\parindent \textindent}
683 \def\narrower{\advance\leftskip\parindent
684 \advance\rightskip\parindent}
685
686 % LaTeX has its own sectioning macros
687 %\outer\def\beginsection#1\par{\vskip\z@ plus.3\vsizel\penalty-250
688 % \vskip\z@ plus-.3\vsizel\bigskip\vskip\parskip
689 % \message{#1}\leftline{\bf#1}\nobreak\smallskip\noindent}
690
691 \outer\def\proclaim #1. #2\par{\medbreak
692 \noindent{\bf#1.\enspace}{\sl#2\par}%
693 \ifdim\lastskip<\medskipamount \remove\lastskip\penalty55\medskip\fi}
694
695 \def\raggedright{\rightskip\z@ plus2em \spaceskip.3333em \xspaceskip.5em\relax}
696 \def\ttraggedright{\tt\rightskip\z@ plus2em\relax} % for use with \tt only
697
698 \chardef\%=\'%
699 \chardef\&=\'&
700 \chardef\#=\'#
701 \chardef\$=\'$
702 \chardef\ss="19
703 \chardef\ae="1A
704 \chardef\oe="1B
705 \chardef\o="1C
706 \chardef\AE="1D
707 \chardef\OE="1E
708 \chardef\O="1F
709 \chardef\i="10 \chardef\j="11 % dotless letters
710 \def\aa{\accent23a}
711 \def\l{\char32l}
712 \def\L{\leavevmode\setbox0\hbox{L}\hbox to\wd0{\hss\char32L}}
713
714 \def\leavevmode{\unhbox\voidb@x} % begins a paragraph, if necessary
715 \def\_ {\leavevmode \kern.06em \vbox{\hrule width.3em}}
716 \def\AA{\leavevmode\setbox0\hbox{h}\dimen@ht\advance\dimen@-1ex%
717 \rlap{\raise.67\dimen@\hbox{\char'27}}A}
718
719 \def\mathhexbox#1#2#3{\leavevmode
720 \hbox{\$m@th \mathchar"#1#2#3$}}
721 \def\dag{\mathhexbox279}
722 \def\ddag{\mathhexbox27A}
723 \def\S{\mathhexbox278}
724 \def\P{\mathhexbox27B}
725
726 \def\oalign#1{\leavevmode\vtop{\baselineskip\z@skip \lineskip.25ex%
727 \ialign{##\crrc#1\crrc}}}% % put characters over each other
728 \def\oalign{\lineskiplimit-\maxdimen \oalign}
729
730 % LaTeX change: |d, |b, |c redefined to work in a moving argument.
731 \def\pd#1{\oalign{#1\crrc\hidewidth.\hidewidth}}
732 \def\d{\protect\pd}
733
734 \def\pb#1{\oalign{#1\crrc\hidewidth
735 \vbox to.2ex{\hbox{\char22}\vss}\hidewidth}}
736 \def\b{\protect\pb}
737
738 \def\pc#1{\setbox\z@\hbox{#1}\ifdim\ht\z@=1ex\accent24 #1%

```

```

739 \else{\oalign{\unhbox\z@\cr\cr\hidewidth\char24\hidewidth}}\fi}
740 \def\c{\protect\pc}
741 % end of LaTeX change to \d, \b, \c
742
743 \def\copyright{\oalign{\hfil\raise.07ex\hbox{c}\hfil\cr\mathhexbox20D}}
744
745 % LaTeX change: \ldots is redefined to do essentially what Plain's \dots does,
746 % so ...
747 \def\dots{\ldots}
748
749 \def\TeX{T\kern-.1667em\lower.5ex\hbox{E}\kern-.125emX}
750
751 \def\`#1{\accent18 #1}
752 \def\'#1{\accent19 #1}
753 \def\~#1{\accent20 #1} \let\^=\v
754 \def\u#1{\accent21 #1} \let\^S=\u
755 \def\=#1{\accent22 #1}
756 \def\^#1{\accent94 #1} \let\^D=\^
757 \def\.#1{\accent95 #1}
758 \def\H#1{\accent"7D #1}
759 \def\~#1{\accent"7E #1}
760 \def\"#1{\accent"7F #1}
761
762 % LaTeX change: Make \t work in a moving argument.
763 \def\pt#1{\edef\next{\the\font}\the\textfont1\accent"7F\next#1}
764 \def\t{\protect\pt}
765
766 % LaTeX change: \kern\z@ added to end of \hrulefill and \dotfill
767 % to make them work in 'tabular' and 'array' environments.
768 % (Change made 24 July 1987).
769 \def\hrulefill{\leaders\hrule\hfill\kern\z@}
770 \def\dotfill{\cleaders\hbox{\m@th \mkern1.5mu.\mkern1.5mu}\hfill\kern\z@}
771
772 \def\rightarrowfill{\m@th\mathord-\mkern-6mu%
773 \cleaders\hbox{\mkern-2mu\mathord-\mkern-2mu}\hfill
774 \mkern-6mu\mathord\rightarrow}
775 \def\leftarrowfill{\m@th\mathord\leftarrow\mkern-6mu%
776 \cleaders\hbox{\mkern-2mu\mathord-\mkern-2mu}\hfill
777 \mkern-6mu\mathord-}
778 \mathchardef\braceld="37A \mathchardef\bracerd="37B
779 \mathchardef\bracelu="37C \mathchardef\braceru="37D
780 \def\downbracefill{\m@th\braceld\leaders\vrule\hfill\braceru
781 \bracelu\leaders\vrule\hfill\bracerd}
782 \def\upbracefill{\m@th\bracelu\leaders\vrule\hfill\bracerd
783 \braceld\leaders\vrule\hfill\braceru}
784
785 % LaTeX change: \bye is eliminated.
786 %\outer\def\bye{\par\vfill\supereject\end}
787
788 % Macros for math setting
789 \message{math definitions,}
790
791 \let\sp=\^ \let\sb=_
792 \def\,\{\mskip\thinmuskip}
793 \def\>\{\mskip\medmuskip}
794 \def\;\{\mskip\thickmuskip}
795 \def\!\{\mskip-\thinmuskip}
796 \def\*{\discretionary{\thinspace\the\textfont2\char2}{}}
797 {\catcode'\=' \active \gdef'\^{\bgroup\prim@s}}
798 \def\prim@s{\prime\futurelet\next\pr@m@s}
799 \def\pr@m@s{\ifx'\next\let\nxt\pr@@@s \else\ifx^\next\let\nxt\pr@@@t
800 \else\let\nxt\egroup\fi\fi \nxt}
801 \def\pr@@@s#1{\prim@s} \def\pr@@@t#1#2{\#2\egroup}
802 {\catcode'\^Z=\active \gdef^^Z{\not=}} % ^^Z is like \ne in math
803
804 {\catcode\'_=\active \global\let=_} % _ in math is either subscript or \_
805
806 \mathchardef\alpha="010B
807 \mathchardef\beta="010C
808 \mathchardef\gamma="010D
809 \mathchardef\delta="010E
810 \mathchardef\epsilon="010F
811 \mathchardef\zeta="0110
812 \mathchardef\eta="0111

```

```

813 \mathchardef\theta="0112
814 \mathchardef\iota="0113
815 \mathchardef\kappa="0114
816 \mathchardef\lambda="0115
817 \mathchardef\mu="0116
818 \mathchardef\nu="0117
819 \mathchardef\xi="0118
820 \mathchardef\pi="0119
821 \mathchardef\rho="011A
822 \mathchardef\sigma="011B
823 \mathchardef\tau="011C
824 \mathchardef\upsilon="011D
825 \mathchardef\phi="011E
826 \mathchardef\chi="011F
827 \mathchardef\psi="0120
828 \mathchardef\omega="0121
829 \mathchardef\varepsilon="0122
830 \mathchardef\vartheta="0123
831 \mathchardef\varpi="0124
832 \mathchardef\varrho="0125
833 \mathchardef\varsigma="0126
834 \mathchardef\varphi="0127
835 \mathchardef\Gamma="7000
836 \mathchardef\Delta="7001
837 \mathchardef\Theta="7002
838 \mathchardef\Lambda="7003
839 \mathchardef\Xi="7004
840 \mathchardef\Pi="7005
841 \mathchardef\Sigma="7006
842 \mathchardef\Upsilon="7007
843 \mathchardef\Phi="7008
844 \mathchardef\Psi="7009
845 \mathchardef\Omega="700A
846
847 \mathchardef\aleph="0240
848 \def\hbar{{\mathchar'26\mkern-9mu}}
849 \mathchardef\imath="017B
850 \mathchardef\jmath="017C
851 \mathchardef\ell="0160
852 \mathchardef\wp="017D
853 \mathchardef\Re="023C
854 \mathchardef\Im="023D
855 \mathchardef\partial="0140
856 \mathchardef\infty="0231
857 \mathchardef\prime="0230
858 \mathchardef\emptyset="023B
859 \mathchardef\nabla="0272
860 \def\surd{{\mathchar"1270}}
861 \mathchardef\top="023E
862 \mathchardef\bot="023F
863 \def\angle{{\vbox{\ialign{\$m@th\scriptstyle##$\crrc
864     \not\mathrel{\mkern14mu}\crrc
865     \noalign{\nointerlineskip}
866     \mkern2.5mu\leaders\hrule height.34pt\hfill\mkern2.5mu\crrc}}}}
867 \mathchardef\triangle="0234
868 \mathchardef\forall="0238
869 \mathchardef\exists="0239
870 \mathchardef\neg="023A \let\not=\neg
871 \mathchardef\flat="015B
872 \mathchardef\natural="015C
873 \mathchardef\sharp="015D
874 \mathchardef\clubsuit="027C
875 \mathchardef\diamondsuit="027D
876 \mathchardef\heartsuit="027E
877 \mathchardef\spadesuit="027F
878
879 \mathchardef\coprod="1360
880 \mathchardef\bigvee="1357
881 \mathchardef\bigwedge="1356
882 \mathchardef\biguplus="1355
883 \mathchardef\bigcap="1354
884 \mathchardef\bigcup="1353
885 \mathchardef\intop="1352 \def\int{\intop\nolimits}
886 \mathchardef\prod="1351

```

```

887 \mathchardef\sum="1350
888 \mathchardef\bigotimes="134E
889 \mathchardef\bigoplus="134C
890 \mathchardef\bigodot="134A
891 \mathchardef\ointop="1348 \def\oint{\ointop\nolimits}
892 \mathchardef\bigsqcup="1346
893 \mathchardef\smallint="1273
894
895 \mathchardef\triangleleft="212F
896 \mathchardef\triangleright="212E
897 \mathchardef\bigtriangleup="2234
898 \mathchardef\bigtriangledown="2235
899 \mathchardef\wedge="225E \let\land=\wedge
900 \mathchardef\vee="225F \let\lor=\vee
901 \mathchardef\cap="225C
902 \mathchardef\cup="225B
903 \mathchardef\ddagger="227A
904 \mathchardef\dagger="2279
905 \mathchardef\sqcap="2275
906 \mathchardef\sqcup="2274
907 \mathchardef\uplus="225D
908 \mathchardef\amalg="2271
909 \mathchardef\diamond="2205
910 \mathchardef\bullet="220F
911 \mathchardef\wr="226F
912 \mathchardef\div="2204
913 \mathchardef\odot="220C
914 \mathchardef\oslash="220B
915 \mathchardef\otimes="220A
916 \mathchardef\ominus="2209
917 \mathchardef\oplus="2208
918 \mathchardef\mp="2207
919 \mathchardef\pm="2206
920 \mathchardef\circ="220E
921 \mathchardef\bigcirc="220D
922 \mathchardef\setminus="226E % for set difference A\setminus B
923 \mathchardef\cdot="2201
924 \mathchardef\ast="2203
925 \mathchardef\times="2202
926 \mathchardef\star="213F
927
928 \mathchardef\propto="322F
929 \mathchardef\sqsubseteq="3276
930 \mathchardef\sqsupseteq="3277
931 \mathchardef\parallel="326B
932 \mathchardef\mid="326A
933 \mathchardef\dashv="3261
934 \mathchardef\vdash="3260
935 \mathchardef\nearrow="3225
936 \mathchardef\searrow="3226
937 \mathchardef\nwarrow="322D
938 \mathchardef\swarrow="322E
939 \mathchardef\Leftrightarrow="322C
940 \mathchardef\Leftarrow="3228
941 \mathchardef\Rightarrow="3229
942 \def\neq{\not=} \let\ne=\neq
943 \mathchardef\leq="3214 \let\le=\leq
944 \mathchardef\geq="3215 \let\ge=\geq
945 \mathchardef\succ="321F
946 \mathchardef\prec="321E
947 \mathchardef\approx="3219
948 \mathchardef\succeq="3217
949 \mathchardef\preceq="3216
950 \mathchardef\supset="321B
951 \mathchardef\subset="321A
952 \mathchardef\supseteq="3213
953 \mathchardef\subseteq="3212
954 \mathchardef\in="3232
955 \mathchardef\ni="3233 \let\owns=\ni
956 \mathchardef\gg="321D
957 \mathchardef\ll="321C
958 \mathchardef\not="3236
959 \mathchardef\leftrightharrow="3224
960 \mathchardef\leftarrow="3220 \let\gets=\leftarrow

```

```

961 \mathchardef\rightarrow="3221 \let\to=\rightarrow
962 \mathchardef\mapstochar="3237 \def\mapsto{\mapstochar\rightarrow}
963 \mathchardef\sim="3218
964 \mathchardef\simeq="3227
965 \mathchardef\perp="323F
966 \mathchardef\equiv="3211
967 \mathchardef\asymp="3210
968 \mathchardef\smile="315E
969 \mathchardef\frown="315F
970 \mathchardef\leftharpoonup="3128
971 \mathchardef\leftharpoondown="3129
972 \mathchardef\rightharpoonup="312A
973 \mathchardef\rightharpoondown="312B
974
975 \def\joinrel{\mathrel{\mkern-3mu}}
976 \def\relbar{\mathrel{\smash-}} % \smash, because - has the same height as +
977 \def\Relbar{\mathrel=}
978 \mathchardef\hook="312C \def\hookrightarrow{\hook\joinrel\rightarrow}
979 \mathchardef\rhook="312D \def\hookleftarrow{\leftarrow\joinrel\rhook}
980 \def\bowtie{\mathrel\triangleright\joinrel\mathrel\triangleleft}
981 \def\models{\mathrel|\joinrel=}
982 \def\Longrightarrow{\Relbar\joinrel\Rightarrow}
983
984 % LaTeX Change: \longrightarrow and \longleftarrow redefined to make
985 % then robust.
986 \def\longrightarrow{\protect@lra}
987 \def@lra{\relbar\joinrel\rightarrow}
988 \def\longleftarrow{\protect@lla}
989 \def@lla{\leftarrow\joinrel\relbar}
990 % End of LaTeX change to \longrightarrow and \longleftarrow
991
992 \def\Longleftarrow{\Leftarrow\joinrel\Relbar}
993 \def\longmapsto{\mapstochar\longrightarrow}
994 \def\longleftrightarrow{\leftarrow\joinrel\rightarrow}
995 \def\Longleftrightarrow{\Leftarrow\joinrel\Rightarrow}
996 \def\iff{\;\Longleftrightarrow\;}
997
998 \mathchardef\ldotp="613A % ldot as a punctuation mark
999 \mathchardef\cdotp="6201 % cdot as a punctuation mark
1000 \mathchardef\colon="603A % colon as a punctuation mark
1001
1002 % LaTeX change: make \ldots work outside math mode too.
1003 \def@ldots{\mathinner{\ldotp\ldotp\ldotp}}
1004 \def\ldots{\protect\pldots}
1005 \def\pldots{\relax\ifmmode@ldots\else\mbox{\$m@th@ldots\,$}\fi}
1006 % End of LaTeX change to \ldots
1007
1008 \def\cdots{\mathinner{\cdotp\cdotp\cdotp}}
1009 \def\vdots{\vbox{\baselineskip4p@ \lineskiplimit\z@
1010 \kern6p@\hbox{.}\hbox{.}\hbox{.}}}
1011 \def\ddots{\mathinner{\mkern1mu\raise7p@\vbox{\kern7p@\hbox{.}}\mkern2mu
1012 \raise4p@\hbox{.}\mkern2mu\raise7p@\hbox{.}\mkern1mu}}
1013
1014 \def\acute{\mathaccent"7013 }
1015 \def\grave{\mathaccent"7012 }
1016 \def\ddot{\mathaccent"707F }
1017 \def\tilde{\mathaccent"707E }
1018 \def\bar{\mathaccent"7016 }
1019 \def\breve{\mathaccent"7015 }
1020 \def\check{\mathaccent"7014 }
1021 \def\hat{\mathaccent"705E }
1022 \def\vec{\mathaccent"017E }
1023 \def\dot{\mathaccent"705F }
1024 \def\widetilde{\mathaccent"0365 }
1025 \def\widehat{\mathaccent"0362 }
1026 \def\overrightarrow#1{\vbox{\m@th\ialign{##\crrc
1027 \rightarrowfill\crrc\noalign{\kern-\p@\nointerlineskip}
1028 \$\hfil\displaystyle{#1}\hfil$\crrc}}}
1029 \def\overleftarrow#1{\vbox{\m@th\ialign{##\crrc
1030 \leftarrowfill\crrc\noalign{\kern-\p@\nointerlineskip}
1031 \$\hfil\displaystyle{#1}\hfil$\crrc}}}
1032 \def\overbrace#1{\mathop{\vbox{\m@th\ialign{##\crrc\noalign{\kern3p@}
1033 \downbracefill\crrc\noalign{\kern3p@\nointerlineskip}
1034 \$\hfil\displaystyle{#1}\hfil$\crrc}}}\limits}

```



```

1035 \def\underbrace#1{\mathop{\vtop{\m@th\ialign{##\crcr
1036     $\hfil\displaystyle{#1}\hfil$\crcr\noalign{\kern3\p@\nointerlineskip}
1037     \upbracefill\crcr\noalign{\kern3\p@}}}\limits}
1038 \def\skew#1#2#3{{#2{#3\mkern#1mu}\mkern-#1mu}{}}
1039
1040 \def\lmoustache{\delimiter"437A340 } % top from (, bottom from )
1041 \def\rmoustache{\delimiter"537B341 } % top from ), bottom from (
1042 \def\lgroup{\delimiter"462833A } % extensible ( with sharper tips
1043 \def\rgroup{\delimiter"562933B } % extensible ) with sharper tips
1044 \def\arrowvert{\delimiter"26A33C } % arrow without arrowheads
1045 \def\Arrowvert{\delimiter"26B33D } % double arrow without arrowheads
1046 \def\bracevert{\delimiter"77C33E } % the vertical bar that extends braces
1047 \def\Vert{\delimiter"26B30D } \let\lVert=\Vert
1048 \def\vert{\delimiter"26A30C }
1049 \def\uparrow{\delimiter"3222378 }
1050 \def\downarrow{\delimiter"3223379 }
1051 \def\updownarrow{\delimiter"326C33F }
1052 \def\Uparrow{\delimiter"322A37E }
1053 \def\Downarrow{\delimiter"322B37F }
1054 \def\Updownarrow{\delimiter"326D377 }
1055 \def\backslash{\delimiter"26E30F } % for double coset G\backslash H
1056 \def\rangle{\delimiter"526930B }
1057 \def\langle{\delimiter"426830A }
1058 \def\rbrace{\delimiter"5267309 } \let\}=rbrace
1059 \def\lbrace{\delimiter"4266308 } \let\{=lbrace
1060 \def\rceil{\delimiter"5265307 }
1061 \def\lceil{\delimiter"4264306 }
1062 \def\rfloor{\delimiter"5263305 }
1063 \def\lfloor{\delimiter"4262304 }
1064
1065 \def\bigl{\mathopen\big}
1066 \def\bigm{\mathrel\big}
1067 \def\bigr{\mathclose\big}
1068 \def\Bigl{\mathopen\Big}
1069 \def\Bigm{\mathrel\Big}
1070 \def\Bigr{\mathclose\Big}
1071 \def\biggl{\mathopen\bigg}
1072 \def\biggm{\mathrel\bigg}
1073 \def\biggr{\mathclose\bigg}
1074 \def\Biggl{\mathopen\Bigg}
1075 \def\Biggm{\mathrel\Bigg}
1076 \def\Biggr{\mathclose\Bigg}
1077 \def\big#1{{\hbox{${\left#1\ vbox to8.5\p@}\right.\n@space$}}}
1078 \def\Big#1{{\hbox{${\left#1\ vbox to11.5\p@}\right.\n@space$}}}
1079 \def\bigg#1{{\hbox{${\left#1\ vbox to14.5\p@}\right.\n@space$}}}
1080 \def\Bigg#1{{\hbox{${\left#1\ vbox to17.5\p@}\right.\n@space$}}}
1081 \def\n@space{\nulldelimiterspace\z@ \m@th}
1082
1083 \def\choose{\atopwithdelims()}
1084 \def\brack{\atopwithdelims[]}
1085 \def\brace{\atopwithdelims\{}}
1086
1087 \def\sqrt{\radical"270370 }
1088
1089 \def\mathpalette#1#2{\mathchoice{#1\displaystyle{#2}}%
1090     {#1\textstyle{#2}}{#1\scriptstyle{#2}}{#1\scriptscriptstyle{#2}}}
1091 \newbox\rootbox
1092 \def\root#1\of{\setbox\rootbox\hbox{${\m@th\scriptscriptstyle{#1}$}
1093     \mathpalette\root@t}
1094 \def\root@t#1#2{\setbox\z@\hbox{${\m@th#1\sqrt{#2}$}
1095     \dimen@ht\z@ \advance\dimen@-\dp\z@
1096     \mkern5mu\raise.6\dimen@\copy\rootbox \mkern-10mu \box\z@}
1097 \newif\ifv@ \newif\ifh@
1098 \def\vphantom{\v@true\h@false\ph@nt}
1099 \def\hphantom{\v@false\h@true\ph@nt}
1100 \def\phantom{\v@true\h@true\ph@nt}
1101 \def\ph@nt{\ifmmode\def\next{\mathpalette\mathph@nt}%
1102     \else\let\next\makeph@nt\fi\next}
1103 \def\makeph@nt#1{\setbox\z@\hbox{#1}\finph@nt}
1104 \def\mathph@nt#1#2{\setbox\z@\hbox{${\m@th#1{#2}$}\finph@nt}
1105 \def\finph@nt{\setbox\tw@null
1106     \ifv@ \ht\tw@\ht\z@ \dp\tw@\dp\z@\fi
1107     \ifh@ \wd\tw@\wd\z@\fi \box\tw@}
1108 \def\mathstrut{\vphantom{}}

```

```

1109 \def\smash{\relax % \relax, in case this comes first in \halign
1110   \ifmmode\def\next{\mathpalette\mathsm@sh}\else\let\next\makesm@sh
1111   \fi\next}
1112 \def\makesm@sh#1{\setbox\z@\hbox{#1}\finsm@sh}
1113 \def\mathsm@sh#1#2{\setbox\z@\hbox{\$m@th#1{#2}$}\finsm@sh}
1114 \def\finsm@sh{\ht\z@\z@ \dp\z@\z@ \box\z@}
1115
1116 \def\cong{\mathrel{\mathpalette\@vereq\sim}} % congruence sign
1117 \def\@vereq#1#2{\lower.5\p@\vbox{\baselineskip\z@skip\lineskip-.5\p@
1118   \ialign{\$m@th#1\hfil##\hfil$\crrc#2\crrc=\crrc}}}}
1119 \def\notin{\mathrel{\mathpalette@cncel\in}}
1120 \def@cncel#1#2{\m@th\oalign{\$hfil#1\mkern1mu/\hfil$\crrc#1#2$}}
1121 \def\rightleftharpoons{\mathrel{\mathpalette\rlh@{}}}}
1122 \def\rlh@#1{\vcenter{\m@th\hbox{\oalign{\raise2pt
1123   \hbox{#1\rightarrowup$}\crrc
1124   $#1\leftarrowdown$}}}}}}
1125 \def\buildrel#1\over#2{\mathrel{\mathop{\kern\z@#2}\limits^{#1}}}
1126 \def\doteq{\buildrel\textstyle.\over=}
1127
1128 \def\log{\mathop{\rm log}\nolimits}
1129 \def\lg{\mathop{\rm lg}\nolimits}
1130 \def\ln{\mathop{\rm ln}\nolimits}
1131 \def\lim{\mathop{\rm lim}}
1132 \def\limsup{\mathop{\rm lim}\sup}
1133 \def\liminf{\mathop{\rm lim}\inf}
1134 \def\sin{\mathop{\rm sin}\nolimits}
1135 \def\arcsin{\mathop{\rm arcsin}\nolimits}
1136 \def\sinh{\mathop{\rm sinh}\nolimits}
1137 \def\cos{\mathop{\rm cos}\nolimits}
1138 \def\arccos{\mathop{\rm arccos}\nolimits}
1139 \def\cosh{\mathop{\rm cosh}\nolimits}
1140 \def\tan{\mathop{\rm tan}\nolimits}
1141 \def\arctan{\mathop{\rm arctan}\nolimits}
1142 \def\tanh{\mathop{\rm tanh}\nolimits}
1143 \def\cot{\mathop{\rm cot}\nolimits}
1144 \def\coth{\mathop{\rm coth}\nolimits}
1145 \def\sec{\mathop{\rm sec}\nolimits}
1146 \def\csc{\mathop{\rm csc}\nolimits}
1147 \def\max{\mathop{\rm max}}
1148 \def\min{\mathop{\rm min}}
1149 \def\sup{\mathop{\rm sup}}
1150 \def\inf{\mathop{\rm inf}}
1151 \def\arg{\mathop{\rm arg}\nolimits}
1152 \def\ker{\mathop{\rm ker}\nolimits}
1153 \def\dim{\mathop{\rm dim}\nolimits}
1154 \def\hom{\mathop{\rm hom}\nolimits}
1155 \def\det{\mathop{\rm det}}
1156 \def\exp{\mathop{\rm exp}\nolimits}
1157 \def\Pr{\mathop{\rm Pr}}
1158 \def\gcd{\mathop{\rm gcd}}
1159 \def\deg{\mathop{\rm deg}\nolimits}
1160
1161 \def\bmod{\mskip-\medmuskip\mkern5mu
1162   \mathbin{\rm mod}\penalty900\mkern5mu\mskip-\medmuskip}
1163 \def\pmod#1{\allowbreak\mkern18mu({\rm mod}\,,\,,#1)}
1164
1165 % RmS 91/08/14 inserted extra braces around entry for NFSS
1166 \def\cases#1{\left\{\,,\vcenter{\normalbaselines\m@th
1167   \ialign{\$##\hfil$&\quad{##}\hfil\crrc#1\crrc}}\right.}
1168 \def\matrix#1{\null\,,\vcenter{\normalbaselines\m@th
1169   \ialign{\hfil$$$&\quad\hfil$$$&\hfil\crrc
1170   \mathstrut\crrc\noalign{\kern-\baselineskip}
1171   #1\crrc\mathstrut\crrc\noalign{\kern-\baselineskip}}}\,,}
1172 \def\pmatrix#1{\left(\matrix{#1}\right)}
1173 \newdimen\p@renwd
1174
1175 % LaTeX: following command is moved after the \tenex font is defined
1176 % by LATEX
1177 %\setbox0=\hbox{\tenex B} \p@renwd=\wd0 % width of the big left (
1178
1179 \def\bordermatrix#1{\begingroup \m@th
1180   \setbox\z@\vbox{\def\cr{\crrc\noalign{\kern2\p@\global\let\cr\endline}}%
1181   \ialign{\$##\hfil\kern2\p@\kern\p@renwd&\thinspace\hfil$$$&\hfil
1182   &\quad\hfil$$$&\hfil\crrc

```

```

1183     \omit\strut\hfil\crrc\noalign{\kern-\baselineskip}%
1184     #1\crrc\omit\strut\cr}}%
1185     \setbox\tw@vbox{\unvcopyz@\global\setbox\@ne\lastbox}%
1186     \setbox\tw@hbox{\unhbox\@ne\unskip\global\setbox\@ne\lastbox}%
1187     \setbox\tw@hbox{\kern\wd\@ne\kern-\p@renwd\left(\kern-\wd\@ne
1188     \global\setbox\@nevbox{\box\@ne\kern2\p@}%
1189     \vcenter{\kern-\ht\@ne\unvboxz@\kern-\baselineskip}\,,\right)$}%
1190     \null\;vbox{\kern\ht\@ne\box\tw@}\endgroup}
1191
1192 \def\openup{\afterassignment\@openup\dimen@=}
1193 \def\@openup{\advance\lineskip\dimen@
1194 \advance\baselineskip\dimen@
1195 \advance\lineskiplimit\dimen@}
1196
1197 % LaTeX change: \equalign eliminated, since it is replaced by the
1198 % eqnarray environment.
1199 %
1200 %\def\equalign#1{\null\,,\vcenter{\openup\jot\m@th
1201 % \ialign{\strut\hfil$displaystyle{##}$\displaystyle{##}$\hfil
1202 % \crrc#1\crrc}}\,,}
1203
1204 \newif\ifdt@p
1205 \def\displ@y{\global\dt@ptrue\openup\jot\m@th
1206 \everycr{\noalign{\ifdt@p \global\dt@pfalse
1207 \vskip-\lineskiplimit \vskip\normallineskiplimit
1208 \else \penalty\interdisplaylinepenalty \fi}}}
1209 \def\@lign{\tabskipz@skip\everycr{}} % restore inside \displ@y
1210 \def\displaylines#1{\displ@y
1211 \halign\hbox to\displaywidth{\@lign\hfil\displaystyle##\hfil}$\crrc
1212 #1\crrc}}
1213
1214 % LaTeX: The following \equalign type macros are eliminated, since
1215 % they are replaced by the eqnarray environment.
1216 %
1217 %\def\equalignno#1{\displ@y \tabskip\centering
1218 % \halign to\displaywidth{\hfil$\@lign\displaystyle{##}$\tabskipz@skip
1219 % $\@lign\displaystyle{##}$\hfil\tabskip\centering
1220 % \llap{\@lign##}$\tabskipz@skip\crrc
1221 % #1\crrc}}
1222 %\def\equalignno#1{\displ@y \tabskip\centering
1223 % \halign to\displaywidth{\hfil$\@lign\displaystyle{##}$\tabskipz@skip
1224 % $\@lign\displaystyle{##}$\hfil\tabskip\centering
1225 % \kern-\displaywidth\rlap{\@lign##}$\tabskip\displaywidth\crrc
1226 % #1\crrc}}
1227
1228 % Definitions related to output
1229
1230 % LaTeX uses its own output routine
1231 %
1232 %\message{output routines,}
1233 %
1234 %\countdef\pageno=0 \pageno=1 % first page is number 1
1235 %\newtoks\headline \headline={\hfil} % headline is normally blank
1236 %\newtoks\footline \footline={\hss\tenrm\folio\hss}
1237 % % footline is normally a centered page number in font \tenrm
1238 %\newif\ifr@ggedbottom
1239 %\def\raggedbottom{\topskip 10\p@ plus60\p@ \r@ggedbottomtrue}
1240 %\def\normalbottom{\topskip 10\p@ \r@ggedbottomfalse} % undoes \raggedbottom
1241 %\def\folio{\ifnum\pageno<z@ \romannumeral-\pageno \else\number\pageno \fi}
1242 %\def\nopagenumbers{\footline{\hfil}} % blank out the footline
1243 %\def\advancepageno{\ifnum\pageno<z@ \global\advance\pageno\m@ne
1244 % \else\global\advance\pageno\@ne \fi} % increase \pageno/
1245
1246 % LaTeX does use the same insert for footnotes as PLAIN
1247 \newinsert\footins
1248
1249 %\def\footnote#1{\let\@sf\empty % parameter #2 (the text) is read later
1250 % \ifhmode\edef\@sf{\spacefactor\the\spacefactor}\fi
1251 % #1\@sf\vfootnote{#1}}
1252 %\def\vfootnote#1{\insert\footins\bgroup
1253 % \interlinepenalty\interfootnotelinepenalty
1254 % \splittopskip\ht\strutbox % top baseline for broken footnotes
1255 % \splitmaxdepth\dp\strutbox \floatingpenalty\@MM
1256 % \leftskipz@skip \rightskipz@skip \spaceskipz@skip \xspaceskipz@skip

```

```

1257 % \textindent{#1}\footstrut\futurelet\next\fo@t}
1258 %\def\fo@t{\ifcat\bgroup\noexpand\next \let\next\fo@t}
1259 % \else\let\next\fo@t\fi \next}
1260 %\def\fo@t{\bgroup\aftergroup\@foot\let\next}
1261 %\def\fo@t#1{#1\@foot}
1262 %\def\@foot{\strut\egroup}
1263 %\def\footstrut{\vbox to\splittopskip{}}
1264
1265 % LaTeX leaves these initializations for the \footins insert.
1266 %
1267 \skip\footins=\bigskipamount % space added when footnote is present
1268 \count\footins=1000 % footnote magnification factor (1 to 1)
1269 \dimen\footins=8in % maximum footnotes per page
1270
1271 %\newinsert\topins
1272 %\newif\ifp@ge \newif\if@mid
1273 %\def\topinsert{\@midfalse\p@gefalse\@ins}
1274 %\def\midinsert{\@midtrue\@ins}
1275 %\def\pageinsert{\@midfalse\p@getrue\@ins}
1276 %\skip\topins=\z@skip % no space added when a topinsert is present
1277 \count\topins=1000 % magnification factor (1 to 1)
1278 \dimen\topins=\maxdimen % no limit per page
1279 %\def\@ins{\par\begingroup\setbox\z@\vbox\bgroup} % start a \vbox
1280 %\def\endinsert{\egroup % finish the \vbox
1281 % \if@mid \dimen@ht\z@ \advance\dimen@dp\z@ \advance\dimen@12\p@
1282 % \advance\dimen@\pagetotal \advance\dimen@-\pageshrink
1283 % \ifdim\dimen@>\pagegoal\@midfalse\p@gefalse\fi\fi
1284 % \if@mid \bigskip\box\z@\bigbreak
1285 % \else\insert\topins{\penalty100 % floating insertion
1286 % \splittopskip\z@skip
1287 % \splitmaxdepth\maxdimen \floatingpenalty\z@
1288 % \ifp@ge \dimen@\dp\z@
1289 % \vbox to\size{\unvbox\z@\kern-\dimen@}% depth is zero
1290 % \else \box\z@\nobreak\bigskip\fi}\fi\endgroup}
1291
1292 %\output{\plainoutput}
1293 %\def\plainoutput{\shipout\vbox{\makeheadline\pagebody\makefootline}}%
1294 % \advancepageno
1295 % \ifnum\outputpenalty>-\@MM \else\dosupereject\fi}
1296 %\def\pagebody{\vbox to\size{\boxmaxdepth\maxdepth \pagecontents}}
1297 %\def\makeheadline{\vbox to\z@{\vskip-22.5\p@
1298 % \line{\vbox to8.5\p@{\the\headline}\vss}\nointerlineskip}
1299 %\def\makefootline{\baselineskip24\p@\line{\the\footline}}
1300 %\def\dosupereject{\ifnum\insertpenalties>\z@ % something is being held over
1301 % \line{\kern-\topskip\nobreak\fill\supereject\fi}
1302 %
1303 %\def\pagecontents{\ifvoid\topins\else\unvbox\topins\fi
1304 % \dimen@=\dp@cclv \unvbox@cclv % open up \box255
1305 % \ifvoid\footins\else % footnote info is present
1306 % \vskip\skip\footins
1307 % \footnoterule
1308 % \unvbox\footins\fi
1309 % \ifr@ggedbottom \kern-\dimen@ \vfil \fi}
1310
1311 % LaTeX keeps PLAIN TeX's \footnoterule as the default
1312 %
1313 \def\footnoterule{\kern-3\p@
1314 \hrule width 2truein \kern 2.6\p@} % the \hrule is .4pt high
1315
1316 % Hyphenation, miscellaneous macros, and initial values for standard layout
1317 \message{hyphenation}
1318
1319 \lefthyphenmin=2 \righthyphenmin=3 % disallow x- or -xx breaks
1320 \input lhypen
1321
1322 % \magnification doesn't work in LaTeX
1323 %
1324 %\def\magnification{\afterassignment\m@g\count@}
1325 %\def\m@g{\mag\count@}
1326 % \hsize6.5truein\vsiz8.9truein\dimen\footins8truein}
1327
1328 %% FMi & RmS 91/08/26: added \errorcontextlines=\maxdimen,
1329 %% suggested by J. Schrod
1330

```

```
1331 \def\tracingall{\tracingcommands\tw@\tracingstats\tw@
1332   \tracingpages\@ne\tracingoutput\@ne\tracinglostchars\@ne
1333   \tracingmacros\tw@\tracingparagraphs\@ne\tracingrestores\@ne
1334   \showboxbreadth\maxdimen\showboxdepth\maxdimen\errorstopmode
1335   \errorcontextlines\maxdimen\tracingonline\@ne}
1336
1337 \def\showhyphens#1{\setbox0\vbox{\parfillskip\z@skip\hsize\maxdimen\tenrm
1338   \pretolerance\m@ne\tolerance\m@ne\hbadness0\showboxdepth0\ #1}}
1339
1340 % input LaTeX fonts and commands
1341
1342 % Initialize \reset@font for use with Old font selection
1343
1344 \let\reset@font\empty
1345
1346 \input lfons
1347 \input latex
1348
1349 % LaTeX change: moved from above.
1350 \setbox0=\hbox{\tenex B} \p@renwd=\wd0 % width of the big left (
1351
1352 % RmS 91/06/21: clear \box0
1353 \setbox0=\box\voidb@x
1354
1355 % LaTeX: default values
1356 \normalbaselines
1357 % LaTeX: select 10pt font size and \rm style
1358 \xpt
1359
1360 \nonfrenchspacing % punctuation affects the spacing
1361 \catcode'\@=12 % at signs are no longer letters
1362
1363 % LaTeX: File called LPLAIN
1364 % Identify the current format
1365 \def\fmtname{lplain}\def\fmtversion{2.09-March 25, 1992}
1366
1367 \typeout{Input any local modifications here.}
1368
1369 \endinput
```

2.2 lhyphen.tex

```
1 % File LHYPHEN - Created 04 November 1991.
2 % Copyright (C) 1991 by Leslie Lamport
3 %
4
5 \message{Loading hyphenation patterns for US english.}
6
7 % PREPARING A FOREIGN LANGUAGE VERSION:
8 %
9 % This file loads hyphenation patterns for US english.
10 % If you want to load additional or other hyphenation patterns,
11 % you have to change this file, which is hereby allowed explicitly,
12 % under the restriction that if you do so, you MUST also change the
13 % message above.
14 %
15
16 \language=0
17 \lefthyphenmin=2 \righthyphenmin=3 % disallow x- or -xx breaks
18 \input hyphen
19
20 \endinput
```

2.3 lfonts.tex

```

1 % File LFONTs - Version of 25 November 1991
2 %
3 % This version of LFONTs.TEX is for the CMR fonts. It was converted
4 % from the AMR version by David Fuchs on 18 December 1985.
5 % And corrected on 11 Nov 1986 by Leslie Lamport.
6 % Last vestige of AMR fonts removed 3 Mar 89 by Leslie Lamport.
7 % Added \endinput 10 Oct 91 by RmS.
8 % Added some \relax in front of \if's 25 Nov 91 by RmS.
9
10 % This file needs to be customized for the fonts available at a particular
11 % site. There are three places where changes need to be made. They
12 % can be found by searching this file for the string FONT-CUSTOMIZING.
13 %
14 % FONT CONVENTIONS
15 %
16 % A TYPESTYLE COMMAND is something like \it that defines a type style.
17 % Each style command \xx is defined to be \protect\pxx, where
18 % \pxx is defined to choose the correct font for the current size.
19 % This allows style commands to appear in 'unsafe' arguments where
20 % protection is required.
21 %
22 % A SIZE COMMAND is something like \normalsize that defines a type size.
23 % It is defined by the document style. However, \normalsize is handled
24 % somewhat differently because it is called so often--e.g., on every
25 % page by the output routine. The document style defines \@normalsize
26 % instead of \normalsize.
27 %
28 % A ONE-SIZE typestyle is one that exists only in the \normalsize size.
29 %
30 % A FONT-SIZE COMMAND is one that defines \textfont, \scriptfont and
31 % \scriptscriptfont for the font families corresponding to preloaded fonts,
32 % as well as the typestyle commands for the preloaded fonts. Each
33 % font-size command has an associated @fontsize command, having the same
34 % name except for an '@' at the front. All font-size commands are defined
35 % in LFONTs. The naming convention is that a fifteenpt font has a font-size
36 % name \xvpt, and so on.
37 %
38 % Each size command \SIZE executes the command
39 % \setsize\SIZE{BASELINESKIP}\FONTSIZE\@FONTSIZE
40 % which does the following.
41 % 0. Executes \@nomath\SIZE to issue warning if in math mode.
42 % 1. \let \currsize = \SIZE
43 % 2. Sets \strutbox to a strut of height .7 * BASELINESKIP and
44 % depth .3 * BASELINESKIP
45 %
46 % Note: Charles Karney observed that step 2 is useless, since the
47 % \FONTSIZE command executed in step 4 resets \strutbox using
48 % the actual baselineskip, which is \baselinestretch * BASELINESKIP.
49 % Some day, this step may get removed. (Note made 28 Feb 89)
50 %
51 % 3. Sets \baselineskip to \baselinestretch * BASELINESKIP
52 % and
53 % 4. Calls \FONTSIZE
54 % 5. Executes the \@FONTSIZE command.
55 % It should then define all the typestyle commands not defined by the font-size
56 % command, except for the one-size type styles. A typestyle command for which
57 % the corresponding font exists but is not preloaded is defined to expand to a
58 % \@getfont command. A typestyle whose font does not exist is defined to
59 % expand to a \@subfont command.
60 %
61 % A one-size typestyle whose font is not preloaded is defined to expand to
62 % a \@onesizefont command.
63 %
64 % \em is defined to be \it inside an unslanted style and \rm inside a
65 % slanted style. An \em command in a section title will produce a \pem
66 % command in the table of contents.
67 %
68
69 \def\em{\protect\pem{}}
70 \def\pem{\relax\ifdim \fontdimen\@ne\font >\z@ \rm \else \it \fi}
71
72 \def\normalsize{\relax\ifx\@currsize\normalsize \rm \else \@normalsize\fi}

```

```

73
74 % \load{SIZE}{STYLE} : Solves anomaly of loaded-on-demand font
75 %   used for first time in math mode. Give this command outside math
76 %   mode, before formula using it for first time.
77 \def\load#1#2{\let\@tempa\@currsize \let\@currsize\relax #1#2\@tempa}
78
79 % \newfont{CMD}{FONT} defines CMD to be the font FONT.
80 %   It is equivalent to \font CMD = FONT
81 % \symbol{NUM} == \char NUM
82
83 \def\newfont#1#2{\@ifdefinable #1{\font #1=#2\relax}}
84 \def\symbol#1{\char #1\relax}
85
86
87
88 % \@getfont STYLE FAM \@FONTSIZE{LOADING.INFO}
89 %   STYLE      = style command
90 %   FAM        = a control sequence defined by \newfam FAM
91 %   \@FONTSIZE = the @fontsize command for the current size.
92 %   LOADING.INFO = information needed to load the font--e.g.,
93 %                 cmtti10 \magstep 2
94 %   Does the following, where FONTNAME denotes a new unique, untypeable
95 %   font name:
96 %     1. Executes \font FONTNAME = LOADING.INFO
97 %     2. Appends '\textfont FAM FONTNAME \defSTYLE{\fam FAM FONTNAME}'
98 %        to the definition of \@FONTSIZE.
99 %     3. Executes \@FONTSIZE STYLE.
100 %
101 % \@nohyphens STYLE \@FONTSIZE
102 %   Used right after \@getfont to set \hyphenchar of the new font to -1,
103 %   thereby prohibiting hyphenation. It is used with \tt fonts.
104 %   (\@nohyphens was added on 12/18/85)
105 %
106 % \@subfont STYLE REPSTYLE
107 %   STYLE, REPSTYLE = type style commands.
108 %   Types warning message and defines uses REPSTYLE.
109 %
110 % \@onesizefont STYLE {LOADING.INFO}
111 %   Defines STYLE to be a typestyle that exists only for the \normalsize
112 %   size. It produces the font specified by LOADING.INFO
113 %
114 % \@addfontinfo \@FONTSIZE{DEFS}
115 %   Expands DEFS and adds to the definition of \@FONTSIZE. Items that should
116 %   not be expanded should be protected with \prct---except no protection
117 %   is needed for '\def\foo', only for the contents of the definition.
118 %
119 % \@nomath CS : Types a warning '\CS used in math mode' if encountered
120 %   in math mode.
121
122 % Remove \outer from definition of \newfam
123 \def\newfam{\alloc@8\fam\chardef\sixt@n}
124
125 \def\@setsize#1#2#3#4{\@nomath#1\let\@currsize#1\baselineskip
126   #2\setbox\strutbox\hbox{\vrule height.7\baselineskip
127     depth.3\baselineskip width\z@}\baselineskip\baselinestretch\baselineskip
128   \normalbaselineskip\baselineskip#3#4}
129
130 \newif\if@bold
131
132 \let\@prtct=\relax
133
134 \def\@addfontinfo#1#2{\def\@prtct{\noexpand\@prtct\noexpand}\def\def{\noexpand
135   \def\noexpand}\xdef#1{#1#2}}
136
137 \def\@getfont#1#2#3#4{\@ifundefined{\string #1\string #3}{\global\expandafter
138   \font \csname \string #1\string #3\endcsname #4\relax
139   \@addfontinfo#3{\textfont #2\csname \string #1\string #3\endcsname
140   \scriptfont #2\csname \string #1\string #3\endcsname
141   \scriptscriptfont #2\csname \string #1\string #3\endcsname
142   \def#1{\fam #2\csname\string #1\string #3\endcsname}}}{#3#1}
143
144 \def\@nohyphens#1#2{\global\expandafter \hyphenchar\csname
145   \string #1\string #2\endcsname \m@ne}
146

```



```

147 \def\@subfont#1#2{\@warning{No \string#1\space typeface in
148   this size, using \string#2}#2}
149
150 \def\@onesizefont#1#2{\expandafter\newfam\csname fm\string#1\endcsname
151   \global\expandafter\font\csname ft\string#1\endcsname #2\relax
152   \gdef#1{\relax\ifx \@currsize\normalsize \@ftfam#1\else
153     \@warning{Typeface \string#1\space available only in
154     \string\normalsize, using \string\rm}\gdef #1{\relax\ifx \@currsize\normalsize
155     \textfont\@fontfam#1 \scriptfont\@fontfam#1 \scriptscriptfont
156     \@fontfam#1\@ftfam#1\else \rm\fi}#1\fi}#1}
157
158 \def\@ftfam#1{\fam\csname fm\string#1\endcsname\csname ft\string#1\endcsname}
159
160 \def\@nomath#1{\relax\ifmmode \@warning{\string#1\space in math mode.}\fi}
161 \def\@nomathbold{\relax\ifmmode
162   \@warning{\string\mathbold\space in math mode.}\fi}
163
164 % The following definitions save token space. E.g., using \@height
165 % instead of height saves 5 tokens at the cost in time of one macro
166 % expansion.
167
168 \def\@height{height}
169 \def\@depth{depth}
170 \def\@width{width}
171
172 \def\@magscale#1{ scaled \magstep #1}
173 \def\@halfmag{ scaled \magstephalf}
174 \def\@ptscale#1{ scaled #100}
175
176
177 %% FONT-CUSTOMIZING: The following \font commands define the
178 %% preloaded LaTeX fonts. Font names should be changed to cause
179 %% different fonts to be loaded in place of these particular AMR fonts.
180 %% \font commands should be added or deleted to change which fonts
181 %% are preloaded.
182
183 % five point
184 \font\fiivrm = cmr5           % roman
185 \font\fiivmi = cmmi5         % math italic
186   \skewchar\fiivmi ='177     % for placement of accents
187 \font\fiivmb = cmmib10 \@ptscale5 % bold math italic
188 \font\fiivsy = cmsy5         % math symbols
189   \skewchar\fiivsy ='60      % for placement of math accents
190 \font\fiivsyb = cmbisy10 \@ptscale5 % bold math symbols
191 \font\fiivit = cmti7 scaled 714 % text italic
192 \font\fiivsl = cmsl10 \@ptscale5 % slanted
193 \font\fiivbf = cmbx7 scaled 714 % extended bold
194 \font\fiivbfs = cmbxsl10\@ptscale5 % extended bold slanted
195 \font\fiivtt = cmtt10 \@ptscale5 % typewriter
196 \font\fiivtti = cmitt10 \@ptscale5 % italic typewriter
197 \font\fiivtts = cmsl10\@ptscale5 % slanted typewriter
198 \font\fiivsf = cmss10 \@ptscale5 % sans serif
199 \font\fiivsfi = cmssi10 \@ptscale5 % italic sans serif
200 \font\fiivsfb = cmssb10\@ptscale5 % bold sans serif
201 \font\fiivsc = cmcsc10 \@ptscale5 % small caps
202 \font\fiivly = lasy5         % LaTeX symbols
203 \font\fiivlyb = lasyb10 \@ptscale5 % LaTeX symbols
204 \font\fiivvit = cmu10 \@ptscale5 % unslanted italic
205
206 % six point
207 \font\siixrm = cmr6           % roman
208 \font\siixmi = cmmi6         % math italic
209   \skewchar\siixmi ='177     % for placement of accents
210 \font\siixmb = cmmib10 \@ptscale6 % bold math italic
211 \font\siixsy = cmsy6         % math symbols
212   \skewchar\siixsy ='60      % for placement of math accents
213 \font\siixsyb = cmbisy10 \@ptscale6 % bold math symbols
214 \font\siixit = cmti7 scaled 857 % text italic
215 \font\siixsl = cmsl10 \@ptscale6 % slanted
216 \font\siixbf = cmbx7 scaled 857 % extended bold
217 \font\siixbfs = cmbxsl10\@ptscale6 % extended bold slanted
218 \font\siixtt = cmtt10 \@ptscale6 % typewriter
219 \font\siixtti = cmitt10 \@ptscale6 % italic typewriter
220 \font\siixtts = cmsl10\@ptscale6 % slanted typewriter

```

```

221 %\font\siasf = cmss10 \@ptscale6 % sans serif
222 %\font\siasfi = cmssi10 \@ptscale6 % italic sans serif
223 %\font\siasfb = cmssbæ10 \@ptscale6 % bold sans serif
224 %\font\siæsc = cmcsc10 \@ptscale6 % small caps
225 \font\sixly = lasy6 % LaTeX symbols
226 %\font\siælyb = lasyb10 \@ptscale6 % LaTeX symbols
227 %\font\siæuit = cmu10 \@ptscale6 % unslanted italic
228
229 % seven point
230 \font\sevrn = cmr7 % roman
231 \font\sevmi = cmmi7 % math italic
232 \skewchar\sevmi ='177 % for placement of accents
233 %\font\sevimib = cmmib10 \@ptscale7 % bold math italic
234 \font\sevsy = cmsy7 % math symbols
235 \skewchar\sevsy ='60 % for placement of math accents
236 %\font\sevsyb = cmbisy10 \@ptscale7 % bold math symbols
237 \font\sevit = cmti7 % text italic
238 %\font\sevsl = cmsl10 \@ptscale7 % slanted
239 %\font\sevbf = cmbæ7 % extended bold
240 %\font\sevbfbs = cmbæsl10 \@ptscale7 % extended bold slanted
241 %\font\sevtt = cmtt10 \@ptscale7 % typewriter
242 %\font\sevtti = cmitt10 \@ptscale7 % italic typewriter
243 %\font\sevttts = cmsl10 \@ptscale7 % slanted typewriter
244 %\font\sevtf = cmss10 \@ptscale7 % sans serif
245 %\font\sevtfi = cmssi10 \@ptscale7 % italic sans serif
246 %\font\sevtfb = cmssbæ10 \@ptscale7 % bold sans serif
247 %\font\sevtfsc = cmcsc10 \@ptscale7 % small caps
248 \font\sevly = lasy7 % LaTeX symbols
249 %\font\sevlyb = lasyb10 \@ptscale7 % LaTeX symbols
250 %\font\sevuit = cmu10 \@ptscale7 % unslanted italic
251
252 % eight point
253 \font\egtrn = cmr8 % roman
254 \font\egtmi = cmmi8 % math italic
255 \skewchar\egtmi ='177 % for placement of accents
256 %\font\egtmib = cmmib10 \@ptscale8 % bold math italic
257 \font\egtsy = cmsy8 % math symbols
258 \skewchar\egtsy ='60 % for placement of math accents
259 %\font\egtsyb = cmbisy10 \@ptscale8 % bold math symbols
260 \font\egtit = cmti8 % text italic
261 %\font\egtisl = cmsl8 % slanted
262 %\font\egtbf = cmbæ8 % extended bold
263 %\font\egtbfbs = cmbæsl10 \@ptscale8 % extended bold slanted
264 %\font\egtitt = cmtt8 % typewriter
265 %\font\egttti = cmitt10 \@ptscale8 % italic typewriter
266 %\font\egttts = cmsl10 \@ptscale8 % slanted typewriter
267 %\font\egtsf = cmss10 \@ptscale8 % sans serif
268 %\font\egtsfi = cmssi10 \@ptscale8 % italic sans serif
269 %\font\egtsfb = cmssbæ10 \@ptscale8 % bold sans serif
270 %\font\egtsfsc = cmcsc10 \@ptscale8 % small caps
271 \font\egtly = lasy8 % LaTeX symbols
272 %\font\egtlyb = lasyb10 \@ptscale8 % LaTeX symbols
273 %\font\egtuit = cmu10 \@ptscale8 % unslanted italic
274
275
276 % nine point
277 \font\ninrm = cmr9 % roman
278 \font\ninmi = cmmi9 % math italic
279 \skewchar\ninmi ='177 % for placement of accents
280 %\font\ninmib = cmmib10 \@ptscale9 % bold math italic
281 \font\ninsy = cmsy9 % math symbols
282 \skewchar\ninsy ='60 % for placement of math accents
283 %\font\ninsyb = cmbisy10 \@ptscale9 % bold math symbols
284 \font\nitit = cmti9 % text italic
285 %\font\ninsl = cmsl9 % slanted
286 \font\ninbf = cmbæ9 % extended bold
287 %\font\ninbfbs = cmbæsl10 \@ptscale9 % extended bold slanted
288 \font\nintt = cmtt9 % typewriter
289 \hyphenchar\nintt = -1 % suppress hyphenation in \tt font
290 %\font\nintti = cmitt10 \@ptscale9 % italic typewriter
291 %\font\nintts = cmsl10 \@ptscale9 % slanted typewriter
292 %\font\ninsf = cmss10 \@ptscale9 % sans serif
293 %\font\ninsfi = cmssi10 \@ptscale9 % italic sans serif
294 %\font\ninsfb = cmssbæ10 \@ptscale9 % bold sans serif

```

```

295 %\font\ninsec = cmcsc10 \@ptscale9 % small caps
296 \font\ninly = lasy9 % LaTeX symbols
297 %\font\ninlyb = lasyb10 \@ptscale9 % LaTeX symbols
298 %\font\ninuit = cmu10 \@ptscale9 % unslanted italic
299
300 % ten point
301 \font\tenrm = cmr10 % roman
302 \font\tenmi = cmmi10 % math italic
303 \skewchar\tenmi ='177 % for placement of accents
304 %\font\tenmib = cmmib10 % bold math italic
305 \font\tensty = cmsy10 % math symbols
306 \skewchar\tensty ='60 % for placement of math accents
307 %\font\tenstyb = cmbsty10 % bold symbols
308 \font\tenit = cmti10 % text italic
309 \font\tenzl = cmsl10 % slanted
310 \font\tenbf = cmbx10 % extended bold
311 %\font\tenzbf = cmbzsl10 % extended bold slanted
312 \font\tentt = cmtt10 % typewriter
313 \hyphenchar\tentt = -1 % suppress hyphenation in \tt font
314 %\font\tentti = cmitt10 % italic typewriter
315 %\font\tentts = cmsliti10 % slanted typewriter
316 \font\tenzsf = cmss10 % sans serif
317 %\font\tenzsf = cmssi10 % italic sans serif
318 %\font\tenzsfb = cmssbx10 % bold sans serif
319 %\font\tenzsc = cmcsc10 % small caps
320 \font\tenly = lasy10 % LaTeX symbols
321 %\font\tenlyb = lasyb10 % bold LaTeX symbols
322 %\font\tenuit = cmu10 % unslanted italic
323
324 % eleven point
325 \font\elvrn = cmr10 \@halfmag % roman
326 \font\elvmi = cmmi10 \@halfmag % math italic
327 \skewchar\elvmi ='177 % for placement of accents
328 %\font\elvmib = cmmib10 \@halfmag % bold math italic
329 \font\elvsy = cmsy10 \@halfmag % math symbols
330 \skewchar\elvsy ='60 % for placement of math accents
331 %\font\elvsyb = cmbsty10 \@halfmag % bold symbols
332 \font\elvit = cmti10 \@halfmag % text italic
333 \font\elvzl = cmsl10 \@halfmag % slanted
334 \font\elvbf = cmbx10 \@halfmag % exeluded bold
335 %\font\elvbfs = cmbzsl10 \@halfmag % exeluded bold slanted
336 \font\elvtl = cmtt10 \@halfmag % typewriter
337 \hyphenchar\elvtl = -1 % suppress hyphenation in \tt font
338 %\font\elvtti = cmitt10 \@halfmag % italic typewriter
339 %\font\elvtts = cmsliti10 \@halfmag % slanted typewriter
340 \font\elzsf = cmss10 \@halfmag % sans serif
341 %\font\elzsf = cmssi10 \@halfmag % italic sans serif
342 %\font\elzsfb = cmssbx10 \@halfmag % bold sans serif
343 %\font\elzsc = cmcsc10 \@halfmag % small caps
344 \font\elvly = lasy10 \@halfmag % LaTeX symbols
345 %\font\elvlyb = lasyb10 \@halfmag % bold LaTeX symbols
346 %\font\elvwit = cmu10 \@halfmag % unslanted italic
347
348 % twelve point
349 \font\twlrm = cmr12 % roman
350 \font\twlmi = cmmi12 % math italic
351 \skewchar\twlmi ='177 % for placement of accents
352 %\font\twlmib = cmmib10 @magscale1 % bold math italic
353 \font\twlsy = cmsy10 @magscale1 % math symbols
354 \skewchar\twlsy ='60 % for placement of math accents
355 %\font\twlsyb = cmbsty10 @magscale1 % bold symbols
356 \font\twlit = cmti12 % text italic
357 \font\twlzl = cmsl12 % slanted
358 \font\twlbf = cmbx12 % extended bold
359 %\font\twlbfs = cmbzsl10 @magscale1 % extended bold slanted
360 \font\twlzl = cmtt12 % typewriter
361 \hyphenchar\twlzl = -1 % suppress hyphenation in \tt font
362 %\font\twlzl = cmitt10 @magscale1 % italic typewriter
363 %\font\twlzl = cmsliti10 @magscale1 % slanted typewriter
364 \font\twzsf = cmss12 % sans serif
365 %\font\twzsf = cmssi10 @magscale1 % italic sans serif
366 %\font\twzsfb = cmssbx10 @magscale1 % bold sans serif
367 %\font\twzsc = cmcsc10 @magscale1 % small caps
368 \font\twlly = lasy10 @magscale1 % LaTeX symbols

```

```

369 %\font\twlllyb = lasyb10 \@magscale1 % bold LaTeX symbols
370 %\font\twluit = cmu10 \@magscale1 % unslanted italic
371
372 % fourteen point
373 \font\frtnrm = cmr10 \@magscale2 % roman
374 \font\frtnmi = cmmi10 \@magscale2 % math italic
375 \skewchar\frtnmi = '177 % for placement of accents
376 %\font\frtnmib = cmmib10 \@magscale2 % bold math italic
377 \font\frtnsy = cmsy10 \@magscale2 % math symbols
378 \skewchar\frtnsy = '60 % for placement of math accents
379 %\font\frtnsyb = cmbsty10 \@magscale2 % bold symbols
380 %\font\frtnit = cmti10 \@magscale2 % text italic
381 %\font\frtnsl = cmsl10 \@magscale2 % slanted
382 \font\frtnbf = cmbx10 \@magscale2 % extended bold
383 %\font\frtnbfs = cmbxsl10 \@magscale2 % extended bold slanted
384 %\font\frnttt = cmtt10 \@magscale2 % typewriter
385 %\font\frnttti = cmitt10 \@magscale2 % italic typewriter
386 %\font\frnttts = cmsl10 \@magscale2 % slanted typewriter
387 %\font\frtnsf = cmss10 \@magscale2 % sans serif
388 %\font\frtnsfi = cmssi10 \@magscale2 % italic sans serif
389 %\font\frtnsfb = cmssbx10 \@magscale2 % bold sans serif
390 %\font\frtnsc = cmcsc10 \@magscale2 % small caps
391 \font\frtnly = lasy10 \@magscale2 % LaTeX symbols
392 %\font\frtnlyb = lasyb10 \@magscale2 % bold LaTeX symbols
393 %\font\frtnuit = cmu10 \@magscale2 % unslanted italic
394
395 % seventeen point
396 \font\svtnrm = cmr17 % roman
397 \font\svtnmi = cmmi10 \@magscale3 % math italic
398 \skewchar\svtnmi = '177 % for placement of accents
399 %\font\svtnmib = cmmib10 \@magscale3 % bold math italic
400 \font\svtnsy = cmsy10 \@magscale3 % math symbols
401 \skewchar\svtnsy = '60 % for placement of math accents
402 %\font\svtnsyb = cmbsty10 \@magscale3 % bold symbols
403 %\font\svtnit = cmti10 \@magscale3 % text italic
404 %\font\svtnsl = cmsl10 \@magscale3 % slanted
405 \font\svtnbf = cmbx10 \@magscale3 % extended bold
406 %\font\svtnbfs = cmbxsl10 \@magscale3 % extended bold slanted
407 %\font\svnttt = cmtt10 \@magscale3 % typewriter
408 %\font\svnttti = cmitt10 \@magscale3 % italic typewriter
409 %\font\svnttts = cmsl10 \@magscale3 % slanted typewriter
410 %\font\svtnsf = cmss10 \@magscale3 % sans serif
411 %\font\svtnsfi = cmssi10 \@magscale3 % italic sans serif
412 %\font\svtnsfb = cmssbx10 \@magscale3 % bold sans serif
413 %\font\svtnsc = cmcsc10 \@magscale3 % small caps
414 \font\svtnly = lasy10 \@magscale3 % LaTeX symbols
415 %\font\svtnlyb = lasyb10 \@magscale3 % bold LaTeX symbols
416 %\font\svtnuit = cmu10 \@magscale3 % unslanted italic
417
418 % twenty point
419 \font\twtyrm = cmr10 \@magscale4 % roman
420 \font\twtyri = cmri10 \@magscale4 % math italic
421 \skewchar\twtyri = '177 % for placement of accents
422 %\font\twtyrib = cmmib10 \@magscale4 % bold math italic
423 \font\twtyry = cmsy10 \@magscale4 % math symbols
424 \skewchar\twtyry = '60 % for placement of math accents
425 %\font\twtyryb = cmbsty10 \@magscale4 % bold symbols
426 %\font\twtyrit = cmti10 \@magscale4 % text italic
427 %\font\twtyrsl = cmsl10 \@magscale4 % slanted
428 %\font\twtyrbf = cmbx10 \@magscale4 % extended bold
429 %\font\twtyrbfs = cmbxsl10 \@magscale4 % extended bold slanted
430 %\font\twtyrtyt = cmtt10 \@magscale4 % typewriter
431 %\font\twtyrtti = cmitt10 \@magscale4 % italic typewriter
432 %\font\twtyrtts = cmsl10 \@magscale4 % slanted typewriter
433 %\font\twtyrsf = cmss10 \@magscale4 % sans serif
434 %\font\twtyrsfi = cmssi10 \@magscale4 % italic sans serif
435 %\font\twtyrsfb = cmssbx10 \@magscale4 % bold sans serif
436 %\font\twtyrsc = cmcsc10 \@magscale4 % small caps
437 \font\twtyry = lasy10 \@magscale4 % LaTeX symbols
438 %\font\twtyryb = lasyb10 \@magscale4 % bold LaTeX symbols
439 %\font\twtyryit = cmu10 \@magscale4 % unslanted italic
440
441 % twenty-five point
442 \font\twfvrm = cmr10 \@magscale5 % roman

```

```

443 %\font\twfumi = cmmi10 \@magscale5 % math italic
444 %\font\twfumib = cmmib10 \@magscale5 % bold math italic
445 %\font\twfusy = cmsy10 \@magscale5 % math symbols
446 %\font\twfusyb = cmbisy10 \@magscale5 % bold symbols
447 %\font\twfvit = cmti10 \@magscale5 % text italic
448 %\font\twfvsl = cmsl10 \@magscale5 % slanted
449 %\font\twfvbf = cmbx10 \@magscale5 % extended bold
450 %\font\twfvbfs = cmbxsl10 \@magscale5 % extended bold slanted
451 %\font\twfvtt = cmtt10 \@magscale5 % typewriter
452 %\font\twfvtti = cmtt10 \@magscale5 % italic typewriter
453 %\font\twfvttt = cmsl10 \@magscale5 % slanted typewriter
454 %\font\twfvst = cmsst10 \@magscale5 % sans serif
455 %\font\twfvstf = cmsst10 \@magscale5 % italic sans serif
456 %\font\twfvstfb = cmsstb10 \@magscale5 % bold sans serif
457 %\font\twfvsc = cmcsc10 \@magscale5 % small caps
458 %\font\twfvly = lasy10 \@magscale5 % LaTeX symbols
459 %\font\twfvlyb = lasyb10 \@magscale5 % bold LaTeX symbols
460 %\font\twfvuit = cmu10 \@magscale5 % unslanted italic
461
462 % Math extension
463 \font\tenex = cmex10
464
465 % line & circle fonts
466 \font\tenln = line10
467 \font\tenlnw = linewidth10
468 \font\tenirc = lcircle10 % 21 Nov 89 : circle10 -> lcircle10
469 \font\tenircw = lcirclew10 % 21 Nov 89 : circlew10 -> lcirclew10
470
471 % Change made 6 May 86: '\@warning' replaced by '\immediate\write 15'
472 % since it's not defined when lfonts.tex is read by lplain.tex.
473 %
474 \ifnum\fontdimen8\tenln=\fontdimen8\tenirc \else
475 \immediate\write 15{Incompatible thin line and circle fonts}\fi
476 \ifnum\fontdimen8\tenlnw=\fontdimen8\tenircw \else
477 \immediate\write 15{Incompatible thick line and circle fonts}\fi
478
479
480 % protected font names
481 \def\rm{\protect\prm}
482 \def\it{\protect\pit}
483 \def\bf{\protect\pbf}
484 \def\sl{\protect\psl}
485 \def\sf{\protect\psf}
486 \def\sc{\protect\psc}
487 \def\tt{\protect\ptt}
488
489 %% FONT-CUSTOMIZING: The following definitions define certain commands
490 %% to be abbreviations for certain font names. These commands are used
491 %% below in \@getfont commands, which load the loaded-on-demand fonts.
492 %% This is done only to save space. To change the fonts that are loaded
493 %% on demand, one can either change these definitions or else change
494 %% the arguments of the \@getfont commands.
495 %%
496 \def\@mbi{cmmib10}
497 \def\@mbsy{cmbisy10}
498 \def\@mcsc{cmcsc10}
499 \def\@mss{cmss10}
500 \def\@lasyb{lasyb10}
501
502 % families
503
504 \newfam\itfam % \it is family 4
505 \newfam\slfam % \sl is family 5
506 \newfam\bfam % \bf is family 6
507 \newfam\ttfam % \tt is family 7
508 \newfam\sffam % \sf is family 8
509 \newfam\scfam % \sc is family 9
510 \newfam\lyfam % \ly is family 10
511
512 \def\cal{\fam\tw@}
513 \def\mit{\fam@ne}
514
515 \def\@setstrut{\setbox\strutbox=\hbox{\vrule \@height .7\baselineskip
516 \@depth .3\baselineskip \@width\z@}}

```

```

517
518
519 %% FONT-CUSTOMIZING: The commands \vpt, \vipt, ... , \xvpt perform all
520 %% the declarations needed to change the type size to 5pt, 6pt, ... ,
521 %% 25pt. To see how this works, consider the definition of \vipt,
522 %% which determines the fonts used in a 7pt type size. The command
523 %% \def\pit{\fam\itfam\sevit}
524 %% means that the \it command causes the preloaded \sevit font to
525 %% be used--this font was defined earlier with a \font\sevit...
526 %% command. The commands
527 %% \textfont\itfam\sevit
528 %% \scriptfont\itfam\sevit
529 %% \scriptscriptfont\itfam\sevit
530 %% tell TeX to use the \sevit font for all three math-mode sizes
531 %% (text, script, and scriptscript) for the 7pt size.
532 %% The fonts appearing in these commands must be preloaded.
533 %%
534 %% The command
535 %% \def\pbf{\@getfont\pbf\bfam\@vipt{ambx7}}
536 %% declares \bf to use a loaded-on-demand font--namely, the font
537 %% ambx7.
538 %%
539 %% The command
540 %% \def\ptt{\@subfont\tt\rm}
541 %% declares that the \tt font is unavailable in the 7pt size, so
542 %% the \rm font is used instead. (The substituted type style should
543 %% correspond to a preloaded size.)
544
545 \def\vpt{\textfont\z@\fivrm
546 \scriptfont\z@\fivrm \scriptscriptfont\z@\fivrm
547 \textfont\@ne\fivmi \scriptfont\@ne\fivmi \scriptscriptfont\@ne\fivmi
548 \textfont\tw@\fivsy \scriptfont\tw@\fivsy \scriptscriptfont\tw@\fivsy
549 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
550 \def\prm{\fam\z@\fivrm}%
551 \def\unboldmath{\everymath{}\everydisplay{}\@nomath
552 \unboldmath\fam\@ne\@boldfalse}\@boldfalse
553 \def\boldmath{\@subfont\boldmath\unboldmath}%
554 \def\pit{\@subfont\it\rm}%
555 \def\psl{\@subfont\sl\rm}%
556 \def\pbf{\@getfont\pbf\bfam\@vpt{cmbx5}}%
557 \def\ptt{\@subfont\tt\rm}%
558 \def\psf{\@subfont\sf\rm}%
559 \def\psc{\@subfont\sc\rm}%
560 \def\ly{\fam\lyfam\fivly}\textfont\lyfam\fivly
561 \scriptfont\lyfam\fivly \scriptscriptfont\lyfam\fivly
562 \@setstrut\rm}
563
564 \def\@vpt{}
565
566 \def\vipt{\textfont\z@\sixrm
567 \scriptfont\z@\sixrm \scriptscriptfont\z@\sixrm
568 \textfont\@ne\sixmi \scriptfont\@ne\sixmi \scriptscriptfont\@ne\sixmi
569 \textfont\tw@\sixsy \scriptfont\tw@\sixsy \scriptscriptfont\tw@\sixsy
570 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
571 \def\prm{\fam\z@\sixrm}%
572 \def\unboldmath{\everymath{}\everydisplay{}\@nomath
573 \unboldmath\@boldfalse}\@boldfalse
574 \def\boldmath{\@subfont\boldmath\unboldmath}%
575 \def\pit{\@subfont\it\rm}%
576 \def\psl{\@subfont\sl\rm}%
577 \def\pbf{\@getfont\pbf\bfam\@vipt{cmbx6}}%
578 \def\ptt{\@subfont\tt\rm}%
579 \def\psf{\@subfont\sf\rm}%
580 \def\psc{\@subfont\sc\rm}%
581 \def\ly{\fam\lyfam\sixly}\textfont\lyfam\sixly
582 \scriptfont\lyfam\sixly \scriptscriptfont\lyfam\sixly
583 \@setstrut\rm}
584
585 \def\@vipt{}
586
587 \def\vipt{\textfont\z@\sevrn
588 \scriptfont\z@\sixrm \scriptscriptfont\z@\fivrm
589 \textfont\@ne\sevmi \scriptfont\@ne\fivmi \scriptscriptfont\@ne\fivmi
590 \textfont\tw@\sevsvy \scriptfont\tw@\fivsy \scriptscriptfont\tw@\fivsy

```

```

591 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
592 \def\prm{\fam\z@\sevrn}%
593 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath
594 \unboldmath\@boldfalse\@boldfalse
595 \def\boldmath{\@subfont\boldmath\unboldmath}%
596 \def\pit{\fam\itfam\sevit}\textfont\itfam\sevit
597 \scriptfont\itfam\sevit \scriptscriptfont\itfam\sevit
598 \def\psl{\@subfont\sl\it}%
599 \def\pbf{\@getfont\pbf\bffam\@viipt{cmbx7}}%
600 \def\ptt{\@subfont\tt\rm}%
601 \def\psf{\@subfont\sf\rm}%
602 \def\psc{\@subfont\sc\rm}%
603 \def\ly{\fam\lyfam\sevly}\textfont\lyfam\sevly
604 \scriptfont\lyfam\fivly \scriptscriptfont\lyfam\fivly
605 \setstrut \rm}
606
607 \def\@viipt{}
608
609 \def\viipt{\textfont\z@\egtrm
610 \scriptfont\z@\sixrm \scriptscriptfont\z@\fivrm
611 \textfont\@ne\egtmi \scriptfont\@ne\sixmi \scriptscriptfont\@ne\fivmi
612 \textfont\tw\egtsy \scriptfont\tw\sixsy \scriptscriptfont\tw\fivsy
613 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
614 \def\prm{\fam\z@\egtrm}%
615 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath
616 \unboldmath\@boldfalse\@boldfalse
617 \def\boldmath{\@subfont\boldmath\unboldmath}%
618 \def\pit{\fam\itfam\egtit}\textfont\itfam\egtit
619 \scriptfont\itfam\sevit \scriptscriptfont\itfam\sevit
620 \def\psl{\@getfont\psl\slfam\@viipt{cmsl8}}%
621 \def\pbf{\@getfont\pbf\bffam\@viipt{cmbx8}}%
622 \def\ptt{\@getfont\ptt\ttfam\@viipt{cmtt8}}\@nohyphens\ptt\@viipt}%
623 \def\psf{\@getfont\psf\sfam\@viipt{cmss8}}%
624 \def\psc{\@getfont\psc\scfam\@viipt{\@mcsc \@ptscale8}}%
625 \def\ly{\fam\lyfam\egtly}\textfont\lyfam\egtly
626 \scriptfont\lyfam\sixly \scriptscriptfont\lyfam\fivly
627 \setstrut \rm}
628
629 \def\@viiipt{}
630
631 \def\ixpt{\textfont\z@\ninrm
632 \scriptfont\z@\sixrm \scriptscriptfont\z@\fivrm
633 \textfont\@ne\ninmi \scriptfont\@ne\sixmi \scriptscriptfont\@ne\fivmi
634 \textfont\tw\ninsy \scriptfont\tw\sixsy \scriptscriptfont\tw\fivsy
635 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
636 \def\prm{\fam\z@\ninrm}%
637 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath\unboldmath
638 \@boldfalse\@boldfalse
639 \def\boldmath{\@subfont\boldmath\unboldmath}%
640 \def\pit{\fam\itfam\ninit}\textfont\itfam\ninit
641 \scriptfont\itfam\sevit \scriptscriptfont\itfam\sevit
642 \def\psl{\@getfont\psl\slfam\@ixpt{cmsl9}}%
643 \def\pbf{\fam\bffam\ninbf}\textfont\bffam\ninbf
644 \scriptfont\bffam\ninbf \scriptscriptfont\bffam\ninbf
645 \def\ptt{\fam\ttfam\nintt}\textfont\ttfam\nintt
646 \scriptfont\ttfam\nintt \scriptscriptfont\ttfam\nintt
647 \def\psf{\@getfont\psf\sfam\@ixpt{cmss9}}%
648 \def\psc{\@getfont\psc\scfam\@ixpt{\@mcsc \@ptscale9}}%
649 \def\ly{\fam\lyfam\ninly}\textfont\lyfam\ninly
650 \scriptfont\lyfam\sixly \scriptscriptfont\lyfam\fivly
651 \setstrut \rm}
652
653 \def\@ixpt{}
654
655 \def\xpt{\textfont\z@\tenrm
656 \scriptfont\z@\sevrn \scriptscriptfont\z@\fivrm
657 \textfont\@ne\tenmi \scriptfont\@ne\sevmi \scriptscriptfont\@ne\fivmi
658 \textfont\tw\tensy \scriptfont\tw\sevsy \scriptscriptfont\tw\fivsy
659 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
660 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath\unboldmath
661 \textfont\@ne\tenmi
662 \textfont\tw\tensy \textfont\lyfam\tenly
663 \@boldfalse\@boldfalse
664 \def\boldmath{\ifundefined{tenmib}{\global\font\tenmib\@mbi

```

```

665 \global\font\tenlyb\@lasyb\relax\@addfontinfo\@xpt
666 \global\font\tenlyb\@lasyb\relax\@addfontinfo\@xpt
667 {\def\boldmath{\everymath{\mit}\everydisplay{\mit}\@prtct\@nomathbold
668 \textfont\@ne\tenmib \textfont\tw\tenlyb
669 \textfont\lyfam\tenlyb \@prtct\@boldtrue}}}\@xpt\boldmath}%
670 \def\prm{\fam\z@tenrm}%
671 \def\pit{\fam\itfam\tenit}\textfont\itfam\tenit \scriptfont\itfam\sevit
672 \scriptscriptfont\itfam\sevit
673 \def\psl{\fam\slfam\tensl}\textfont\slfam\tensl
674 \scriptfont\slfam\tensl \scriptscriptfont\slfam\tensl
675 \def\pbf{\fam\bffam\tenbf}\textfont\bffam\tenbf
676 \scriptfont\bffam\tenbf \scriptscriptfont\bffam\tenbf
677 \def\ptt{\fam\ttfam\tentt}\textfont\ttfam\tentt
678 \scriptfont\ttfam\tentt \scriptscriptfont\ttfam\tentt
679 \def\psf{\fam\sffam\tenmf}\textfont\sffam\tenmf
680 \scriptfont\sffam\tenmf \scriptscriptfont\sffam\tenmf
681 \def\psc{\@getfont\psc\scfam\@xpt{\@mcsc}}%
682 \def\ly{\fam\lyfam\tenly}\textfont\lyfam\tenly
683 \scriptfont\lyfam\sevly \scriptscriptfont\lyfam\sevly
684 \@setstrut \rm}
685
686 \def\@xpt{}
687
688 \def\xipt{\textfont\z@\elvrn
689 \scriptfont\z@\egtrm \scriptscriptfont\z@\sixrm
690 \textfont\@ne\elvmi \scriptfont\@ne\egtmi \scriptscriptfont\@ne\sixmi
691 \textfont\tw\elvly \scriptfont\tw\egtly \scriptscriptfont\tw\sixly
692 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
693 \def\unboldmath{\everymath{\everydisplay{\@nomath\unboldmath
694 \textfont\@ne\elvmi \textfont\tw\elvly
695 \textfont\lyfam\elvly \@boldfalse}\@boldfalse
696 \def\boldmath{\@ifundefined{elvmb}{\global\font\elvmb\@mbi\@halfmag
697 \global\font\elvlyb\@lasyb\@halfmag\relax\@addfontinfo\@xipt
698 \global\font\elvlyb\@lasyb\@halfmag\relax\@addfontinfo\@xipt
699 {\def\boldmath{\everymath{\mit}\everydisplay{\mit}\@prtct\@nomathbold
700 \textfont\@ne\elvmb \textfont\tw\elvlyb
701 \textfont\lyfam\elvlyb\@prtct\@boldtrue}}}\@xipt\boldmath}%
702 \def\prm{\fam\z@\elvrn}%
703 \def\pit{\fam\itfam\elvit}\textfont\itfam\elvit
704 \scriptfont\itfam\egtit \scriptscriptfont\itfam\sevit
705 \def\psl{\fam\slfam\elvsl}\textfont\slfam\elvsl
706 \scriptfont\slfam\tensl \scriptscriptfont\slfam\tensl
707 \def\pbf{\fam\bffam\elvbf}\textfont\bffam\elvbf
708 \scriptfont\bffam\elvbf \scriptscriptfont\bffam\elvbf
709 \def\ptt{\fam\ttfam\elvt}\textfont\ttfam\elvt
710 \scriptfont\ttfam\elvt \scriptscriptfont\ttfam\elvt
711 \def\psf{\fam\sffam\elvsf}\textfont\sffam\elvsf
712 \scriptfont\sffam\elvsf \scriptscriptfont\sffam\elvsf
713 \def\psc{\@getfont\psc\scfam\@xipt{\@mcsc\@halfmag}}%
714 \def\ly{\fam\lyfam\elvly}\textfont\lyfam\elvly
715 \scriptfont\lyfam\egtly \scriptscriptfont\lyfam\egtly
716 \@setstrut \rm}
717
718 \def\@xipt{}
719
720 \def\xiip{\textfont\z@\twlrm
721 \scriptfont\z@\egtrm \scriptscriptfont\z@\sixrm
722 \textfont\@ne\twlmi \scriptfont\@ne\egtmi \scriptscriptfont\@ne\sixmi
723 \textfont\tw\twlly \scriptfont\tw\egtly \scriptscriptfont\tw\sixly
724 \textfont\thr@@\tenex \scriptfont\thr@@\tenex \scriptscriptfont\thr@@\tenex
725 \def\unboldmath{\everymath{\everydisplay{\@nomath\unboldmath
726 \textfont\@ne\twlmi
727 \textfont\tw\twlly \textfont\lyfam\twlly
728 \@boldfalse}\@boldfalse
729 \def\boldmath{\@ifundefined{twlmb}{\global\font\twlmb\@mbi\@magscale1\global
730 \font\twllyb\@lasyb\@magscale1\global\font
731 \twllyb\@lasyb\@magscale1\relax\@addfontinfo\@xiip
732 {\def\boldmath{\everymath
733 {\mit}\everydisplay{\mit}\@prtct\@nomathbold
734 \textfont\@ne\twlmb \textfont\tw\twllyb
735 \textfont\lyfam\twllyb\@prtct\@boldtrue}}}\@xiip\boldmath}%
736 \def\prm{\fam\z@\twlrm}%
737 \def\pit{\fam\itfam\twlit}\textfont\itfam\twlit \scriptfont\itfam\egtit
738 \scriptscriptfont\itfam\sevit

```



```

739 \def\psl{\fam\slfam\twlsl}\textfont\slfam\twlsl
740 \scriptfont\slfam\ tensl \scriptscriptfont\slfam\ tensl
741 \def\pbf{\fam\bffam\twlbf}\textfont\bffam\twlbf
742 \scriptfont\bffam\ ninbf \scriptscriptfont\bffam\ ninbf
743 \def\ptt{\fam\ttfam\twl tt}\textfont\ttfam\twl tt
744 \scriptfont\ttfam\ nintt \scriptscriptfont\ttfam\ nintt
745 \def\psf{\fam\sffam\twlsf}\textfont\sffam\twlsf
746 \scriptfont\sffam\ tensf \scriptscriptfont\sffam\ tensf
747 \def\psc{\@getfont\psc\scfam\@xiipt{\@mcsc\@magscale1}}%
748 \def\ly{\fam\lyfam\twlly}\textfont\lyfam\twlly
749 \scriptfont\lyfam\egtly \scriptscriptfont\lyfam\ sixly
750 \@setstrut \rm}
751
752 \def\@xiipt{}
753
754 \def\xivpt{\textfont\z@\frtnrm
755 \scriptfont\z@\tenrm \scriptscriptfont\z@\sevrn
756 \textfont\@ne\frtnmi \scriptfont\@ne\tenmi \scriptscriptfont\@ne\sevni
757 \textfont\tw@\frtnsy \scriptfont\tw@\tensy \scriptscriptfont\tw@\sevsy
758 \textfont\thr@\tenex \scriptfont\thr@\tenex \scriptscriptfont\thr@\tenex
759 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath\unboldmath
760 \textfont\@ne\frtnmi \textfont\tw@\frtnsy
761 \textfont\lyfam\frtnly \@boldfalse}\@boldfalse
762 \def\boldmath{\@ifundefined{frtnmib}{\global\font
763 \frtnmib\@mbi\@magscale2\global\font\frtnsyb\@mbsy\@magscale2
764 \global\font\frtnlyb\@lasyb\@magscale2\relax\@addfontinfo\@xivpt
765 {\def\boldmath{\everymath
766 {\mit}\everydisplay{\mit}}\@prtct\@nomathbold
767 \textfont\@ne\frtnmib \textfont\tw@\frtnsyb
768 \textfont\lyfam\frtnlyb\@prtct\@boldtrue}}}{\@xivpt\boldmath}}%
769 \def\prm{\fam\z@\frtnrm}%
770 \def\pit{\@getfont\pit\itfam\@xivpt{cmti10\@magscale2}}%
771 \def\psl{\@getfont\psl\slfam\@xivpt{cmsl10\@magscale2}}%
772 \def\pbf{\fam\bffam\frtnbf}\textfont\bffam\frtnbf
773 \scriptfont\bffam\tenbf \scriptscriptfont\bffam\ ninbf
774 \def\ptt{\@getfont\ptt\ttfam\@xivpt{cmtt10\@magscale2}\@nohyphens\ptt\@xivpt}%
775 \def\psf{\@getfont\psf\sffam\@xivpt{\@mss\@magscale2}}%
776 \def\psc{\@getfont\psc\scfam\@xivpt{\@mcsc\@magscale2}}%
777 \def\ly{\fam\lyfam\frtnly}\textfont\lyfam\frtnly
778 \scriptfont\lyfam\tenly \scriptscriptfont\lyfam\sevly
779 \@setstrut \rm}
780
781 \def\@xivpt{}
782
783 \def\xviipt{\textfont\z@\svtnrm
784 \scriptfont\z@\twlrm \scriptscriptfont\z@\tenrm
785 \textfont\@ne\svtnmi \scriptfont\@ne\twlmi \scriptscriptfont\@ne\tenmi
786 \textfont\tw@\svtnsy \scriptfont\tw@\twlsy \scriptscriptfont\tw@\tensy
787 \textfont\thr@\tenex \scriptfont\thr@\tenex \scriptscriptfont\thr@\tenex
788 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath\unboldmath
789 \textfont\@ne\svtnmi \textfont\tw@\svtnsy \textfont\lyfam\svtnly
790 \@boldfalse}\@boldfalse
791 \def\boldmath{\@subfont\boldmath\unboldmath}%
792 \def\prm{\fam\z@\svtnrm}%
793 \def\pit{\@getfont\pit\itfam\@xviipt{cmti10\@magscale3}}%
794 \def\psl{\@getfont\psl\slfam\@xviipt{cmsl10\@magscale3}}%
795 \def\pbf{\fam\bffam\svtnbf}\textfont\bffam\svtnbf
796 \scriptfont\bffam\twlbf \scriptscriptfont\bffam\tenbf
797 \def\ptt{\@getfont\ptt\ttfam\@xviipt{cmtt10\@magscale3}\@nohyphens
798 \ptt\@xviipt}%
799 \def\psf{\@getfont\psf\sffam\@xviipt{cmss17}}%
800 \def\psc{\@getfont\psc\scfam\@xviipt{\@mcsc\@magscale3}}%
801 \def\ly{\fam\lyfam\svtnly}\textfont\lyfam\svtnly
802 \scriptfont\lyfam\twlly \scriptscriptfont\lyfam\tenly
803 \@setstrut \rm}
804
805 \def\@xviipt{}
806
807 \def\xxpt{\textfont\z@\twtyrm
808 \scriptfont\z@\frtnrm \scriptscriptfont\z@\twlrm
809 \textfont\@ne\twtymi \scriptfont\@ne\frtnmi \scriptscriptfont\@ne\twlmi
810 \textfont\tw@\twty sy \scriptfont\tw@\frtnsy \scriptscriptfont\tw@\twlsy
811 \textfont\thr@\tenex \scriptfont\thr@\tenex \scriptscriptfont\thr@\tenex
812 \def\unboldmath{\everymath{}\everydisplay{}}\@nomath\unboldmath

```

```

813     \textfont\@ne\twtymi \textfont\tw@twtyisy \textfont\lyfam\twtyly
814     \boldfalse}\@boldfalse
815 \def\boldmath{\@subfont\boldmath\unboldmath}%
816 \def\prm{\fam\z@twtyrm}%
817 \def\pit{\@getfont\pit\itfam\@xxpt{cmti10\@magscale4}}%
818 \def\psl{\@getfont\psl\slfam\@xxpt{cmsl10\@magscale4}}%
819 \def\pbf{\@getfont\pbf\bffam\@xxpt{cmbx10\@magscale4}}%
820 \def\ptt{\@getfont\ptt\ttfam\@xxpt{cmtt10\@magscale4}\@nohyphens\ptt\@xxpt}%
821 \def\psf{\@getfont\psf\sfam\@xxpt{\@mss\@magscale4}}%
822 \def\psc{\@getfont\psc\scfam\@xxpt{\@mcs\@magscale4}}%
823 \def\ly{\fam\lyfam\twtyly}\textfont\lyfam\twtyly
824     \scriptfont\lyfam\frtnly \scriptscriptfont\lyfam\twlly
825 \setstrut \rm}
826
827 \def\@xxpt{}
828
829 \def\@xxvpt{\textfont\z@twfvrm
830     \scriptfont\z@twtyrm \scriptscriptfont\z@svtnrm
831     \textfont\@ne\twtymi \scriptfont\@ne\twtymi \scriptscriptfont\@ne\svtnmi
832     \textfont\tw@twtyisy \scriptfont\tw@twtyisy \scriptscriptfont\tw@svtnsy
833     \textfont\thr@tenex \scriptfont\thr@tenex \scriptscriptfont\thr@tenex}
834 \def\unboldmath{\everymath{\everydisplay}\@nomath\unboldmath
835     \textfont\@ne\twtymi \textfont\tw@twtyisy \textfont\lyfam\twtyly
836     \boldfalse}\@boldfalse
837 \def\boldmath{\@subfont\boldmath\unboldmath}%
838 \def\prm{\fam\z@twfvrm}%
839 \def\pit{\@subfont\it\rm}%
840 \def\psl{\@subfont\sl\rm}%
841 \def\pbf{\@getfont\pbf\bffam\@xxvpt{cmbx10\@magscale5}}%
842 \def\ptt{\@subfont\tt\rm}%
843 \def\psf{\@subfont\sf\rm}%
844 \def\psc{\@subfont\sc\rm}%
845 \def\ly{\fam\lyfam\twtyly}\textfont\lyfam\twtyly
846     \scriptfont\lyfam\twtyly \scriptscriptfont\lyfam\svtnly
847 \setstrut \rm}
848
849 \def\@xxvpt{}
850
851 % SPECIAL LaTeX character definitions
852
853 % Definitions of math operators added by LaTeX
854 \mathchardef\mho"0A30
855 \mathchardef\Join"3A31
856 \mathchardef\Box"0A32
857 \mathchardef\Diamond"0A33
858 \mathchardef\leadsto"3A3B
859 \mathchardef\sqssubset"3A3C
860 \mathchardef\sqsupset"3A3D
861 \def\lhd{\mathbin{< \hbox to -.43em{\hbox{\vrule
862     \@width .065em \@height .55em \@depth .05em}\hbox to .2em{}}}
863 \def\rhd{\mathbin{\hbox to .3em{\hbox{\vrule \@width .065em \@height
864     .55em \@depth .05em}\hbox to -.43em{>}}}
865 \def\unlhd{\mathbin{\leq \hbox to -.43em{\hbox
866     {\vrule \@width .065em \@height .63em \@depth -.08em}\hbox to .2em{}}}
867 \def\unrhd{\mathbin{\hbox to .3em{\hbox
868     {\vrule \@width .065em \@height .63em \@depth -.08em}\hbox to -.43em{\geq}}}
869
870 % Definition of \$ to work in italic font (since it produces a pound sterling
871 % sign in the cmit font.
872
873 \def\${\protect\pdollar}
874 \def\pdollar{\ifdim \fontdimen\@ne\font >\z@ \sl \fi\char'\$}
875
876 % Definition of pound sterling sign.
877 % Modified 10 Apr 89 to work in math mode.
878
879 \def\pounds{\protect\ppounds}
880 \def\ppounds{\relax\ifmode\mathchar"424\else{\it \char'44}\fi}
881
882
883 % Definition of \copyright changed so it works in other type styles,
884 % and so it is robust
885 \def\copyright{\protect\pcopyright}
886 \def\pcopyright{\rm\oalign{\hfil

```

```
887     \raise.07ex\hbox{c}\hfil\crr\mathhexbox20D}}}  
888  
889 \endinput
```

2.4 latex.tex

```

1 % LATEX VERSION 2.09 <25 March 1992>
2 % Copyright (C) 1992 by Leslie Lamport
3
4 \everyjob{\typeout{LaTeX Version 2.09 <25 March 1992>}}
5 \immediate\write10{LaTeX Version 2.09 <25 March 1992>}
6
7 %          TABLE OF CONTENTS
8 % COMMAND LIST ..... 2
9 % GENERAL CONVENTIONS ..... 6
10 % COUNTERS, ETC. .... 7
11 % USEFUL HACKS ..... 8
12 % ERROR HANDLING ..... 12
13 % \par AND \everypar ..... 15
14 % SPACING / LINE AND PAGE BREAKING ..... 17
15 % PROGRAM CONTROL STRUCTURE MACROS ..... 21
16 % FILE HANDLING ..... 24
17 % ENVIRONMENT COUNTER MACROS ..... 27
18 % PAGE NUMBERING ..... 30
19 % CROSS REFERENCING MACROS ..... 31
20 % ENVIRONMENTS ..... 33
21 % MATH ENVIRONMENTS ..... 36
22 % CENTER, FLUSHRIGHT, FLUSHLEFT, ETC. .... 39
23 % VERBATIM ..... 40
24 % THE LIST ENVIRONMENT ..... 41
25 % ITEMIZE AND ENUMERATE ..... 49
26 % BOXES ..... 51
27 % THE TABBING ENVIRONMENT ..... 57
28 % ARRAY AND TABULAR ENVIRONMENTS ..... 63
29 % THE PICTURE ENVIRONMENT ..... 72
30 % THEOREM ENVIRONMENTS ..... 86
31 % LENGTHS ..... 88
32 % THE TITLE ..... 89
33 % SECTIONING ..... 90
34 % TABLE OF CONTENTS, ETC. .... 94
35 % INDEX COMMANDS ..... 97
36 % BIBLIOGRAPHY ..... 98
37 % FLOATS ..... 100
38 % FOOTNOTES ..... 106
39 % INITIAL DECLARATION COMMANDS ..... 110
40 % OUTPUT ..... 113
41 % DEBUGGING AND TEST INITIALIZATIONS ..... 137
42
43
44 \catcode'\~ =13 \def~{\penalty\@M \ }
45
46
47 % *****
48 % *          COMMAND LIST          *
49 % *****
50 %
51 % DECLARATIONS:
52 % PREAMBLE: \nofiles \documentstyle \includeonly
53 %           \makeindex \makeglossary
54 % IN DOCUMENT :
55 %   FONT SELECTION:
56 %     SIZE: \normalsize \small \footnotesize \scriptsize \tiny
57 %           \large \Large \LARGE \huge \Huge
58 %     STYLE: \bf \it \rm \sl \ss \tt \mit[math mode only]
59 %   STYLE:
60 %     PAGE: [all global] \pagestyle \thispagestyle \pagenumbering \head
61 %     MISC: \raggedright \thicklines \thinlines
62 %     PARAMETER: \setlength \settoheight \addtolength \setcounter \addtocounter
63 %     NEW: \newlength \newtheorem \newcommand
64 %     MISC: \savebox \sbox \obeycr \restorecr
65 %
66 % ENVIRONMENTS:
67 % ? -> PAR: document
68 % PAR -> PAR: list enumerate itemize description
69 %           center flushright flushleft
70 %           verbatim picture float
71 % PAR -> BOX: tabular tabbing
72 % PAR -> MATH: math displaymath equation

```

```

73 % MATH -> MATH: array
74 % ANY -> PAR: minipage
75 % ANY -> BOX: stack
76 %
77 % TEXT-PRODUCING:
78 % WITH TEXT ARGUMENT:
79 % ANY -> BOX: \makebox \mbox \framebox \fbox \dashbox
80 % \shortstack \footnotemark \cite[] \raisebox
81 % ANY -> PAR: \parbox[inner]
82 % PAR -> PAR: \chapter \section ... \footnote \footnotetext
83 % \topnewpage \verb
84 % MATH: \sqrt \underline \overline
85 % PICTURE: \put \multiput
86 % LIST: \item
87 % WITHOUT TEXT ARGUMENT:
88 % ANY MODE:
89 % SYMBOLS: \$ \{ \} \_ \@ \& \#
90 % ACCENTS: See TeXbook
91 % OTHER: \rule \ref \pageref \today \usebox \typein \input \cite
92 % MATH: \over
93 % PAR MODE: \include \bibliography \tableofcontents \listoffigures ...
94 % LIST: \item \arabic \roman \Roman \alph \Alph
95 % PICTURE: \line \vector \circle \oval
96 % ARRAY & TABULAR: \hline \vline
97 %
98 % SPACING & BREAKING:
99 % ANY : \hfill \hspace
100 % PAR : \newpage \newpage \vspace \noindent
101 % PAR & INNER MATH
102 % : \newpage \clearpage \cleardoublepage
103 % : \pagebreak \nopagebreak \linebreak \nolinebreak \newline
104 % MATH : \over \; \, \!
105 % MULTILINE : \\
106 % TABBING : \pushtab \poptab \> \< \+ \- \kill ...
107 % ARRAY & TABULAR
108 % : \multicolumn \noalign
109 %
110 % NO DIRECT CHANGES TO DOCUMENT:
111 % \index \glossary \typeout \label \tableentry \stop \protect
112 %
113 % PARAMETERS:
114 %
115 % \columnsep \skip\footin \intextsep
116 % \columnseprule \oddsidemargin
117 % \columnwidth \textfloatsep
118 % \evensidemargin \footsep \textheight
119 % \floatsep \headheight \textwidth
120 % \headsep \topmargin
121 %
122 %
123 % ALPHABETIZED LIST:
124 %
125 % ORDINARY COMMANDS:
126 %
127 % \Alph \include \parbox
128 % \Roman \index \put
129 % \l \item \raisebox
130 % \alph \label \ref
131 % \appendix \line \roman
132 % \arabic \linebreak \rule
133 % \bibliography \listoffigures \section
134 % \chapter \listoftables \shortstack
135 % \circle \makebox \stop
136 % \cite \mbox \subsection
137 % \cite \multicolumn \subsubsection
138 % \cleardoublepage \multiput \tableentry
139 % \clearpage \tableofcontents
140 % \dashbox \newline \today
141 % \fbox \newpage \typein
142 % \footnotemark \noindent \typeout
143 % \footnotetext \nolinebreak \usebox
144 % \framebox \nopagebreak \vector
145 % \glossary \oval \vline
146 % \hline \pagebreak \vspace

```

```

147 % \hspace          \pageref          \protect
148 %
149 %
150 % ENVIRONMENTS & DECLARATIONS:
151 %
152 % For each of these commands, the same command name prefixed by 'end'
153 % is also reserved--e.g., \enddocument.
154 %
155 % \BIG              \footnotesize    \pagestyle
156 % \Big             \head              \picture
157 % \addtocounter    \includeonly    \raggedright
158 % \addtolength    \itemize        \restorecr
159 % \array           \list              \savebox
160 % \big             \makeglossary   \sbox
161 % \center          \makeindex      \scriptscriptsize
162 % \description    \math           \scriptsize
163 % \displaymath    \minipage       \setcounter
164 % \document        \newcommand     \setlength
165 % \documentstyle  \newlength      \settowidth
166 % \enumerate      \newtheorem     \small
167 % \equation        \nofiles        \shortstack\tabbing
168 %                 \normalsize      \tabular
169 % \float           \obeycr         \thicklines
170 % \flushleft      \pagelayout     \thinlines
171 % \flushright     \pagenumbering  \thispagestyle
172 %
173 %
174 % PARAMETERS :
175 %
176 % \columnsep       \footinsertskip \intextsep
177 % \columnseprule   \oddsidemargin
178 % \columnwidth     \textfloatsep
179 % \evensidemargin  \footsep        \textheight
180 % \floatsep        \headheight     \textwidth
181 % \headsep         \topmargin
182 %
183 %
184 % TABBING COMMANDS:
185 %
186 % These commannds are defined only within a tabbing environment.
187 %
188 % \kill           |>   |-
189 % \pushtab       |<   |=
190 % \poptab        |+
191 %
192 %
193 % COMPLETE LIST :
194 % Below is a complete list of every command starting with '\' that
195 % appears in LATEX.
196 %
197 % \
198 % \!
199 % \#
200 % \$
201 % \&
202 % \'
203 % \{
204 % \}
205 % |+
206 % \,
207 % \-
208 % \.
209 % \:
210 % \;
211 % \<
212 % \=
213 % \>
214 % \@
215 % @@
216 % @@end
217 % @@endpbox
218 % @@eqnocr
219 % @@hyph
220 % @@input

```

```
221 % \@par
222 % \@sqrt
223 % \@startpbox
224 % \@underline
225 % \@warning
226 % \@acci
227 % \@accii
228 % \@acciii
229 % \@acol
230 % \@acolampacol
231 % \@addamp
232 % \@addfield
233 % \@addmarginpar
234 % \@addtobot
235 % \@addtocurcol
236 % \@addtodblcol
237 % \@addtoneatcol
238 % \@addtopreamble
239 % \@addtoreset
240 % \@addtotoporbit
241 % \@afterheading
242 % \@afterindentfalse
243 % \@afterindenttrue
244 % \@Alph
245 % \@alph
246 % \@ampacol
247 % \@arabic
248 % \@argarraycr
249 % \@argdef
250 % \@argrbox
251 % \@argtabularcr
252 % \@array
253 % \@arrayacol
254 % \@arrayclassiv
255 % \@arrayclassv
256 % \@arrayclassz
257 % \@arraycr
258 % \@arrayparboxrestore
259 % \@arrayrule
260 % \@arstrut
261 % \@arstrutbox
262 % \@auxout
263 % \@badcrrr
264 % \@badend
265 % \@badlinearg
266 % \@badmath
267 % \@badpoptabs
268 % \@badtab
269 % \@beginparpenalty
270 % \@begintheorem
271 % \@bibitem
272 % \@biblabel
273 % \@bitor
274 % \@botlist
275 % \@botnum
276 % \@botroom
277 % \@bsphack
278 % \@caption
279 % \@capttype
280 % \@car
281 % \@carcube
282 % \@ccclv
283 % \@cdr
284 % \@centercr
285 % \@centering
286 % \@cfla
287 % \@cflb
288 % \@charlb
289 % \@charrb
290 % \@chclass
291 % \@checkend
292 % \@chnum
293 % \@circ
294 % \@circle
```

```
295 % \@circlefnt
296 % \@cite
297 % \@citea
298 % \@citeb
299 % \@citea
300 % \@cla      % counter used in \cline
301 % \@classi
302 % \@classii
303 % \@classiii
304 % \@classiv
305 % \@classv
306 % \@classz
307 % \@clb      % counter used in \cline
308 % \@cline
309 % \@clnht
310 % \@clnwd
311 % \@clubpenalty
312 % \@colht
313 % \@colnum
314 % \@colroom
315 % \@combinedblfloats
316 % \@combinefloats
317 % \@comdblflleft
318 % \@comfleleft
319 % \@cons
320 % \@contfield
321 % \@ctrerr
322 % \@curfield
323 % \@curline
324 % \@currbox
325 % \@currentlabel
326 % \@currentreference
327 % \@currenvir
328 % \@currlist
329 % \@currtype
330 % \@curtab
331 % \@curtabmar
332 % \@dascnt
333 % \@dashbox
334 % \@dashcnt
335 % \@dashdim
336 % \@dblarg
337 % \@dbldeferlist
338 % \@dblfloat
339 % \@dblfloatplacement
340 % \@dblfloatsep
341 % \@dblfpbot
342 % \@dblfpsep
343 % \@dblfpptop
344 % \@dblmaxsep
345 % \@dbltextfloatsep
346 % \@dbltoplist
347 % \@dbltopnum
348 % \@dbltoproom
349 % \@deferlist
350 % \@defincounter
351 % \@defpar
352 % \@depth
353 % \@dischyph
354 % \@docclearpage
355 % \@documentstyle
356 % \@doendpe
357 % \@donoparitem
358 % \@dot
359 % \@dotsep
360 % \@dottedtocline
361 % \@downline
362 % \@downvector
363 % \@eha
364 % \@ehb
365 % \@ehc
366 % \@ehd
367 % \@elt
368 % \@empty
```



```
369 % \@endparenu
370 % \@endparpenalty
371 % \@endpbox
372 % \@endpefalse
373 % \@endpetrue
374 % \@endtabbing
375 % \@endtheorem
376 % \@enumctr
377 % \@enumdepth
378 % \@enumspacing
379 % \@eqnrcr
380 % \@eqnum
381 % \@eqnrel
382 % \@eqnswtrue
383 % \@esphack
384 % \@Esphack
385 % \@evenfoot
386 % \@evenhead
387 % \@expast
388 % \@failedlist
389 % \@fcolmadefalse
390 % \@filesw
391 % \@fileswfalse
392 % \@fileswtrue
393 % \@firstampfalse
394 % \@firstamptrue
395 % \@firstcolumntrue
396 % \@firsttab
397 % \@flfail
398 % \@float
399 % \@floatpenalty
400 % \@floatplacement
401 % \@floatsep
402 % \@flsucceed
403 % \@fltouf
404 % \@flushglue
405 % \@fnsymbol
406 % \@footnotemark
407 % \@footnotetext
408 % \@for
409 % \@forloop
410 % \@fornoop
411 % \@fpbot
412 % \@fpmin
413 % \@fpsep
414 % \@fptop
415 % \@framebox
416 % \@framepicbox
417 % \@freelist
418 % \@getcirc
419 % \@getlarrow
420 % \@getlinechar
421 % \@getpen
422 % \@getrarrow
423 % \@glossaryfile
424 % \@gobble
425 % \@gobblecr
426 % \@gobbletwo
427 % \@gtempa
428 % \@halfwidth
429 % \@halignto
430 % \@hangfrom
431 % \@height
432 % \@highpenalty
433 % \@hightab
434 % \@hline
435 % \@holdpg
436 % \@hspace
437 % \@hspacer
438 % \@hvvector
439 % \@icentercr
440 % \@iden
441 % \@ifatmargin
442 % \@ifdefinable
```

```

443 % \@ifnch
444 % \@ifnextchar
445 % \@iforloop
446 % \@ifframebox
447 % \@ifframepicbox
448 % \@ifstar
449 % \@ifundefined
450 % \@input           % used in \input
451 % \@irsbox
452 % \@makebox
453 % \@makepicbox
454 % \@minipage
455 % \@index
456 % \@indexfile
457 % \@inlabelfalse
458 % \@input
459 % \@inputcheck
460 % \@insertfalse
461 % \@inserttrue
462 % \@iparbox
463 % \@irsbox
464 % \@isavebox
465 % \@isavepicbox
466 % \@ishortstack
467 % \@istackcr
468 % \@itabcr
469 % \@item
470 % \@itemdepth
471 % \@itemfudge
472 % \@itemitem
473 % \@itemlabel
474 % \@itempenalty
475 % \@itemspacing
476 % \@iwhiledim
477 % \@iwhilenum
478 % \@iwhilesw
479 % \@ixstackcr
480 % \@killglue
481 % \@labels
482 % \@lastchclass
483 % \@lateabug
484 % \@lateaerr
485 % \@lbibitem
486 % \@leftcolumn
487 % \@leftmarginsskip
488 % \@leftmark
489 % \@thead
490 % \@linechar
491 % \@linefnt
492 % \@linelen
493 % \@list
494 % \@listctr
495 % \@listdepth
496 % \@listi
497 % \@listii
498 % \@listvi
499 % \@lnbk
500 % \@lowpenalty
501 % \@lquote
502 % \@ltab
503 % \@M
504 % \@m
505 % \@mainaux
506 % \@mainout
507 % \@makebox
508 % \@makecaption
509 % \@makecol
510 % \@makefcolumn
511 % \@makefnmark
512 % \@makefnstext
513 % \@makeonecolumn
514 % \@makeother
515 % \@makepicbox
516 % \@maketwocolumn

```

```
517 % \@marbox
518 % \@markright
519 % \@maxdepth
520 % \@maxsep
521 % \@maxtab
522 % \@medpenalty
523 % \@Mi
524 % \@midlist
525 % \@Mii
526 % \@Miii
527 % \@minipagefalse
528 % \@minipagerestore
529 % \@Miv
530 % \@mkboth
531 % \@mklab
532 % \@mkpream
533 % \@MM
534 % \@mparbottom
535 % \@mparswitchfalse
536 % \@mpfn
537 % \@mpfnnumber
538 % \@mpfootins
539 % \@mpfootnotetext
540 % \@mplistdepth
541 % \@multicnt
542 % \@namedef
543 % \@nameuse
544 % \@nbitem
545 % \@ne
546 % \@negargfalse
547 % \@negargtrue
548 % \@newctr
549 % \@newenv
550 % \@newline
551 % \@newlist
552 % \@newlistfalse
553 % \@next
554 % \@nextchar
555 % \@nextwhile
556 % \@nil
557 % \@nibrlistfalse
558 % \@nibrlisttrue
559 % \@nnil
560 % \@nobreakfalse
561 % \@nocnterr
562 % \@nodocument
563 % \@nofonterror
564 % \@noitemargfalse
565 % \@noitemargtrue
566 % \@noitemerr
567 % \@noligs
568 % \@nolnbk
569 % \@nolnerr
570 % \@noparitemfalse
571 % \@noparitemtrue
572 % \@noparlistfalse
573 % \@noparlisttrue
574 % \@nopgbk
575 % \@normalcr
576 % \@normalsize
577 % \@noskipsecfalse
578 % \@notdefinable
579 % \@notprerr
580 % \@nthm
581 % \@nxttabmar
582 % \@oddfoot
583 % \@oddhead
584 % \@opargbegintheorem
585 % \@opcol
586 % \@optionfiles
587 % \@optionlist
588 % \@options
589 % \@othm
590 % \@outerparskip
```

```

591 % \outputbox
592 % \outputdblcol
593 % \outputpage
594 % \oval
595 % \oubtrue
596 % \ovdx
597 % \ovdy
598 % \ovhorz
599 % \oultrue
600 % \ovri
601 % \ovro
602 % \ovrtrue
603 % \outtrue
604 % \ouvert
605 % \ovxx
606 % \ovy
607 % \pagedp
608 % \pageht
609 % \par
610 % \parboxrestore
611 % \parmoderr
612 % \partaux
613 % \partlist
614 % \partout
615 % \partsw
616 % \partswfalse
617 % \partswtrue
618 % \pboxswfalse
619 % \pboxswtrue
620 % \pgbk
621 % \picbox
622 % \picht
623 % \picture
624 % \pnumwidth
625 % \preamble
626 % \preamblecmds
627 % \preamerr
628 % \put
629 % \qend
630 % \qrelax
631 % \reargdef
632 % \renewenv
633 % \restorepar
634 % \reversemarginfalse
635 % \reversemargintrue
636 % \rhead
637 % \rightmark
638 % \rightskip
639 % \Roman
640 % \Rroman
641 % \rsbox
642 % \rtab
643 % \rule
644 % \sanitize
645 % \savebox
646 % \savemarbox
647 % \savepicbox
648 % \savs
649 % \savsk
650 % \scolelt
651 % \sdblcolelt
652 % \secpenalty
653 % \sect
654 % \setpar
655 % \settab
656 % \sharp
657 % \shortstack
658 % \sline
659 % \spaces
660 % \specialoutput
661 % \specialpagefalse
662 % \specialstyle
663 % \sptoken
664 % \sqrt

```

```
665 % \@ssect
666 % \@startcolumn
667 % \@startdblcolumn
668 % \@startfield
669 % \@startline
670 % \@startpbox
671 % \@startsection
672 % \@starttoc
673 % \@stopfield
674 % \@stopline
675 % \@stpelt
676 % \@svector
677 % \@sverb
678 % \@svsec
679 % \@susechd
680 % \@tabacol
681 % \@tabarray
682 % \@tabclassiv
683 % \@tabclassz
684 % \@tabcr
685 % \@tablab
686 % \@tabminus
687 % \@tabplus
688 % \@tabpush
689 % \@tabrj
690 % \@tabular
691 % \@tabularcr
692 % \@temp
693 % \@tempa
694 % \@tempb
695 % \@tempbox
696 % \@tempboxa
697 % \@tempc
698 % \@tempcnta
699 % \@tempcntb
700 % \@tempd
701 % \@tempdima
702 % \@tempdimb
703 % \@tempe
704 % \@tempskipa
705 % \@tempskipb
706 % \@tempswa
707 % \@tempswafalse
708 % \@tempswatrue
709 % \@temptokena
710 % \@testdef
711 % \@testfp
712 % \@testpach
713 % \@textbottom
714 % \@textfloatsep
715 % \@textmin
716 % \@texttop
717 % \@tfor
718 % \@tforloop
719 % \@thanks
720 % \@thefnmark
721 % \@thefoot
722 % \@thehead
723 % \@themargin
724 % \@themark
725 % \@thm
726 % \@thmcounter
727 % \@thmcountersep
728 % \@toCRMarg
729 % \@toodeep
730 % \@toplist
731 % \@topnewpage
732 % \@topnum
733 % \@toproom
734 % \@topsep
735 % \@topsepadd
736 % \@totalleftmargin
737 % \@trivlist
738 % \@tryfcolumn
```

```
739 % \@trylist
740 % \@twocolumnfalse
741 % \@twoside
742 % \@twosidefalse
743 % \@typein
744 % \@upline
745 % \@upordown
746 % \@upvector
747 % \@verb
748 % \@verbatim
749 % \@vline
750 % \@vobeyspaces
751 % \@vspace
752 % \@vspacer
753 % \@vtryfc
754 % \@vvector
755 % \@warning
756 % \@wckptelt
757 % \@whiledim
758 % \@whilenoop
759 % \@whilenum
760 % \@whilesw
761 % \@whileswnoop
762 % \@wholewidth
763 % \@width
764 % \@wrindex
765 % \@writeckpt
766 % \@writefile
767 % \@wtryfc
768 % \@x@sf
769 % \@xarg
770 % \@xargarraycr
771 % \@xarraycr
772 % \@xbitor
773 % \@xcentercr
774 % \@xdblarg
775 % \@xdblfloat
776 % \@xdim
777 % \@xeqncr
778 % \@xeenooop
779 % \@xexpast
780 % \@xfloat
781 % \@xfootnote
782 % \@xfootnotemark
783 % \@xfootnotenext
784 % \@xhead
785 % \@xifnch
786 % \@xmpar
787 % \@xnewline
788 % \@xnthm
789 % \@xobeysp
790 % \@xsect
791 % \@xstartcol
792 % \@xtabcr
793 % \@xtabularcr
794 % \@xthm
795 % \@xtryfc
796 % \@xtypein
797 % \@xverbatim
798 % \@xaxii
799 % \@xympar
800 % \@yarg
801 % \@yargarraycr
802 % \@ydim
803 % \@yegncr
804 % \@yhead
805 % \@ympar
806 % \@ynthm
807 % \@ythm
808 % \@ytryfc
809 % \@yyarg
810 % \@ztryfc
811 % \@a
812 % \@active
```

```
813 % \addcontentsline
814 % \addpenalty
815 % \addtocontents
816 % \addtocounter
817 % \addtolength
818 % \addvspace
819 % \advance
820 % \alloc@
821 % \allocationnumber
822 % \Alph
823 % \alph
824 % \and
825 % \appendix
826 % \arabic
827 % \array
828 % \arraycolsep
829 % \arrayrulewidth
830 % \arraystretch
831 % \author
832 % \bar
833 % \baselineskip
834 % \begin
835 % \begingroup
836 % \bf
837 % \bgroup
838 % \bibcite
839 % \bibdata
840 % \bibitem
841 % \bibliography
842 % \bibliographystyle
843 % \bibstyle
844 % \BIG
845 % \Big
846 % \big
847 % \bigskip
848 % \botfigrule
849 % \botmark
850 % \botnum
851 % \bottomfraction
852 % \box
853 % \boxmaxdepth
854 % \buildrel
855 % \bullet
856 % \c@bottomnumber
857 % \c@chapter
858 % \c@dbltopnumber
859 % \c@equation
860 % \c@eval
861 % \c@footnote
862 % \c@mpfootnote
863 % \c@page
864 % \c@secnumdepth
865 % \c@section
866 % \c@tocdepth
867 % \c@topnumber
868 % \c@totalnumber
869 % \caption
870 % \catcode
871 % \catcoded
872 % \center
873 % \centering
874 % \chapter
875 % \chaptermark
876 % \char
877 % \chardef
878 % \circle
879 % \cite
880 % \cl@ckpt
881 % \cleardoublepage
882 % \clearpage
883 % \cline
884 % \closeout
885 % \clubpenalty
886 % \columnsep
```

```
887 % \columnseprule
888 % \columnwidth
889 % \contentsline
890 % \copy
891 % \count
892 % \countdef
893 % \cr
894 % \crrc
895 % \csname
896 % \dag
897 % \dagger
898 % \dashbox
899 % \date
900 % \dblfigrule
901 % \dblfloatpagefraction
902 % \dblfloatsep
903 % \dbltextfloatsep
904 % \dbltopfraction
905 % \ddagger
906 % \deadcycles
907 % \def
908 % \description
909 % \dimen
910 % \dimen@
911 % \discretionary
912 % \displaymath
913 % \displaystyle
914 % \displaywidth
915 % \divide
916 % \do
917 % \document
918 % \documentstyle
919 % \dospecials
920 % \doublerulesep
921 % \dp
922 % \edef
923 % \egroup
924 % \else
925 % \end
926 % \end@dblfloat
927 % \end@float
928 % \endarray
929 % \endcsname
930 % \enddocument
931 % \endenumerate
932 % \endequation
933 % \endfigure
934 % \endgroup
935 % \enditemize
936 % \endlist
937 % \endpicture
938 % \endsloppypar
939 % \endtabbing
940 % \endtabular
941 % \endthebibliography
942 % \endtrivlist
943 % \enumerate
944 % \eqnarray
945 % \eqno
946 % \equation
947 % \errmessage
948 % \errorstopmode
949 % \eval
950 % \evensidemargin
951 % \everyjob
952 % \everypar
953 % \expandafter
954 % \extracolsep
955 % \fbox
956 % \fboxrule
957 % \fboxsep
958 % \fi
959 % \figure
```



```
961 % \fill
962 % \firstmark
963 % \float
964 % \floatingpenalty
965 % \floatpagefraction
966 % \floatsep
967 % \flushbottom
968 % \flushleft
969 % \flushright
970 % \fnsymbol
971 % \footins
972 % \footinsertskip
973 % \footnote
974 % \footnotemark
975 % \footnoterule
976 % \footnotesep
977 % \footnotesize
978 % \footnotetext
979 % \footsep
980 % \footskip
981 % \frac
982 % \frame
983 % \framebox
984 % \frenchspacing
985 % \fussy
986 % \futurelet
987 % \gdef
988 % \global
989 % \glossary
990 % \halfwidth
991 % \halign
992 % \hangindent
993 % \hbox
994 % \head
995 % \headheight
996 % \headsep
997 % \hfil
998 % \hfill
999 % \hfuzz
1000 % \hline
1001 % \hrule
1002 % \hsize
1003 % \hskip
1004 % \hspace
1005 % \hss
1006 % \ht
1007 % \Huge
1008 % \huge
1009 % \hyphenchar
1010 % \if
1011 % \if@afterindent
1012 % \if@eqnsw
1013 % \if@endpe
1014 % \if@fcolmade
1015 % \if@filesw
1016 % \if@firstamp
1017 % \if@firstcolumn
1018 % \if@ignore
1019 % \if@inlabel
1020 % \if@insert
1021 % \if@minipage
1022 % \if@mparswitch
1023 % \if@negarg
1024 % \if@newlist
1025 % \if@nmbriest
1026 % \if@nobreak
1027 % \if@noitemarg
1028 % \if@noparitem
1029 % \if@noparlist
1030 % \if@noskipsec
1031 % \if@ovb
1032 % \if@owl
1033 % \if@ovr
1034 % \if@out
```

```
1035 % \if@pboxsw
1036 % \if@reversemargin
1037 % \if@rjfield
1038 % \if@specialpage
1039 % \if@tempwa
1040 % \if@test
1041 % \if@twocolumn
1042 % \if@twoside
1043 % \ifcase
1044 % \ifdim
1045 % \ifeof
1046 % \ifhmode
1047 % \ifinner
1048 % \ifmmode
1049 % \ifnum
1050 % \ifodd
1051 % \ifvmode
1052 % \ifvoid
1053 % \ifx
1054 % \ignorespaces
1055 % \immediate
1056 % \include
1057 % \includeonly
1058 % \indent
1059 % \index
1060 % \indexentry
1061 % \input
1062 % \insecunt
1063 % \insert
1064 % \interdisplaylinepenalty
1065 % \interfootnotelinepenalty
1066 % \interlinepenalty
1067 % \intextsep
1068 % \it
1069 % \item
1070 % \itemindent
1071 % \itemize
1072 % \itemsep
1073 % \jobname
1074 % \kern
1075 % \kill
1076 % \label
1077 % \labelenumi
1078 % \labelenumiv
1079 % \labelitemi
1080 % \labelitemii
1081 % \labelitemiii
1082 % \labelitemiv
1083 % \labelsep
1084 % \labelwidth
1085 % \LARGE
1086 % \Large
1087 % \large
1088 % \lastbox
1089 % \lastskip
1090 % \LaTeX
1091 % \lbrace
1092 % \leaders
1093 % \leavevmode
1094 % \lefteqn
1095 % \leftmargin
1096 % \leftmargini
1097 % \leftmarginiv
1098 % \leftmark
1099 % \leftskip
1100 % \let
1101 % \limits
1102 % \line
1103 % \linebreak
1104 % \lineskip
1105 % \linethickness
1106 % \linewidth
1107 % \list
1108 % \listoffigures
```

```
1109 % \listoftables
1110 % \listparindent
1111 % \llap
1112 % \long
1113 % \lower
1114 % \m@ne
1115 % \m@th
1116 % \makeatletter
1117 % \makeatother
1118 % \makebox
1119 % \makeglossary
1120 % \makeindex
1121 % \makelabel
1122 % \maketitle
1123 % \marginpar
1124 % \marginparpush
1125 % \marginparsep
1126 % \marginparwidth
1127 % \mark
1128 % \markboth
1129 % \markright
1130 % \math
1131 % \mathchar
1132 % \mathchardef
1133 % \mathop
1134 % \mathrel
1135 % \maxdeadcycles
1136 % \maxdepth
1137 % \maxdimen
1138 % \mb@b
1139 % \mb@eval
1140 % \mb@l
1141 % \mb@r
1142 % \mb@t
1143 % \mbox
1144 % \medskip
1145 % \message
1146 % \minipage
1147 % \mit
1148 % \mkern
1149 % \moveright
1150 % \mskip
1151 % \multicolumn
1152 % \multiply
1153 % \multput
1154 % \multispan
1155 % \newbox
1156 % \newcommand
1157 % \newcount
1158 % \newcounter
1159 % \newdimen
1160 % \newenvironment
1161 % \newif
1162 % \newinsert
1163 % \newlabel
1164 % \newlength
1165 % \newline
1166 % \newlinechar
1167 % \newpage
1168 % \newsavebox
1169 % \newskip
1170 % \newswitch
1171 % \newtheorem
1172 % \newtoks
1173 % \newwrite
1174 % \noalign
1175 % \nobreak
1176 % \nocite
1177 % \noexpand
1178 % \nofiles
1179 % \noindent
1180 % \nointerlineskip
1181 % \nolinebreak
1182 % \nonumber
```

```
1183 % \nopagebreak
1184 % \normalbaselineskip
1185 % \normallineskip
1186 % \normalmarginpar
1187 % \normalsize
1188 % \nullfont
1189 % \number
1190 % \numberline
1191 % \obeycr
1192 % \obeylines
1193 % \obeyspaces
1194 % \oddsidemargin
1195 % \of
1196 % \on@line
1197 % \onecolumn
1198 % \openin
1199 % \or
1200 % \outer
1201 % \output
1202 % \outputpenalty
1203 % \oval
1204 % \over
1205 % \overfullrule
1206 % \overline
1207 % \p@
1208 % \pagebreak
1209 % \pagelayout
1210 % \pagenumbering
1211 % \pageref
1212 % \pagestyle
1213 % \par
1214 % \paragraph
1215 % \parbox
1216 % \parfillskip
1217 % \parindent
1218 % \parsep
1219 % \parshape
1220 % \parskip
1221 % \partopsep
1222 % \partsw
1223 % \penalty
1224 % \picture
1225 % \poptab
1226 % \poptabs
1227 % \postdisplaypenalty
1228 % \preveddepth
1229 % \protect
1230 % \ps@empty
1231 % \ps@plain
1232 % \pushtab
1233 % \pushtabs
1234 % \put
1235 % \quotation
1236 % \raggedbottom
1237 % \raggedleft
1238 % \raggedright
1239 % \raise
1240 % \raisebox
1241 % \rbrace
1242 % \read
1243 % \ref
1244 % \refstepcounter
1245 % \relax
1246 % \renewcommand
1247 % \renewenvironment
1248 % \reset@font
1249 % \restorecr
1250 % \reversemarginpar
1251 % \right
1252 % \rightmargin
1253 % \rightmark
1254 % \rightskip
1255 % \rlap
1256 % \rm
```

```
1257 % \Roman
1258 % \roman
1259 % \romannumeral
1260 % \root
1261 % \rule
1262 % \samepage
1263 % \savebox
1264 % \sbox
1265 % \sc
1266 % \scriptscriptsize
1267 % \scriptsize
1268 % \secdef
1269 % \section
1270 % \sectionmark
1271 % \setbox
1272 % \setcounter
1273 % \setlength
1274 % \settowidth
1275 % \shipout
1276 % \shortstack
1277 % \showboxbreadth
1278 % \showboxdepth
1279 % \sift@n
1280 % \skip
1281 % \sl
1282 % \SLiTeX
1283 % \sloppy
1284 % \sloppypar
1285 % \small
1286 % \smallskip
1287 % \space
1288 % \spacefactor
1289 % \splitmaxdepth
1290 % \splittopskip
1291 % \sqrt
1292 % \ss
1293 % \stackrel
1294 % \stepcounter
1295 % \stop
1296 % \stretch
1297 % \string
1298 % \strut
1299 % \subsection
1300 % \subsubsection
1301 % \tabalign
1302 % \tabbing
1303 % \tabbingsep
1304 % \tabcolsep
1305 % \tableentry
1306 % \tableofcontents
1307 % \tabskip
1308 % \tabular
1309 % \tencirc
1310 % \tencircw
1311 % \tenln
1312 % \tenlnw
1313 % \textfloatsep
1314 % \textfraction
1315 % \textheight
1316 % \textwidth
1317 % \thanks
1318 % \the
1319 % \thebibliography
1320 % \theenumi
1321 % \theenumii
1322 % \theequation
1323 % \thefigure
1324 % \thefootnote
1325 % \thempfn
1326 % \thempfootnote
1327 % \thepage
1328 % \thesection
1329 % \thicklines
1330 % \thinlines
```

```

1331 % \thinspace
1332 % \thispagestyle
1333 % \tiny
1334 % \title
1335 % \today
1336 % \tolerance
1337 % \topfigrule
1338 % \topfraction
1339 % \topmargin
1340 % \topnewpage
1341 % \topnum
1342 % \topsep
1343 % \topskip
1344 % \tracingonline
1345 % \tracingoutput
1346 % \tracingstats
1347 % \trivlist
1348 % \tt
1349 % \tw@
1350 % \twocolumn
1351 % \typein
1352 % \typeout
1353 % \unbox
1354 % \underline
1355 % \unhbox
1356 % \unitlength
1357 % \unskip
1358 % \unvbox
1359 % \usebox
1360 % \usecounter
1361 % \vadjust
1362 % \value
1363 % \vbox
1364 % \vcenter
1365 % \vector
1366 % \verb
1367 % \verbatim
1368 % \vfil
1369 % \vfuzz
1370 % \vline
1371 % \vrule
1372 % \vsize
1373 % \vskip
1374 % \vspace
1375 % \vsplit
1376 % \vss
1377 % \vtop
1378 % \wd
1379 % \write
1380 % \writes
1381 % \xdef
1382 % \z@
1383 % \[
1384 % \l
1385 % \]
1386 % \^
1387 % \_
1388 % \`
1389 % \{
1390 % \|
1391 % \}
1392 % \~
1393
1394
1395
1396
1397 % *****
1398 % *          GENERAL CONVENTIONS          *
1399 % *****
1400 %
1401 % THE \LaTeX AND \SLiTeX LOGOS ARE DEFINED HERE.
1402 %
1403 %% RmS 91/09/29: \reset@font added to \LaTeX logo.
1404 \def\p@LaTeX{\reset@font\rm L\kern-.36em\raise.3ex\hbox{\sc a}\kern-.15em%

```

```

1405     T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}}
1406
1407 %% RmS 91/09/29: \SLiTeX logo added.
1408 \def\p@SLiTeX{\reset@font\rm S\kern-.06em{\sc l\kern-.035emi}\kern-.06emT\kern
1409   -.1667em\lower.7ex\hbox{E}\kern-.125emX}}
1410
1411 %% RmS 91/10/17: \protect'ed the logos
1412 \def\LaTeX{\protect\p@LaTeX}
1413 \def\SLiTeX{\protect\p@SLiTeX}
1414
1415
1416 % SAVED VERSIONS OF TeX PRIMITIVES:
1417 %
1418 % The TeX primitive \foo is saved as \@foo . The following primitives
1419 % are handled in this way:
1420
1421 \let\@par=\par
1422 %\let\@relax=\relax % This was needed at one time, but seems to be obsolete.
1423 \let\@input=\input
1424 \let\@end=\end
1425
1426 % The following was added 19 April 1986:
1427 % The \- command is redefined to allow it to work in the \tt type style,
1428 % where automatic hyphenation is suppressed by setting \hyphenchar to -1.
1429 % The original definition is saved as \@hyph just in case anyone needs it.
1430
1431 \let\@hyph=- % Original defin
1432 \def\-\{\discretionary{-}{-}{-}}
1433
1434 % SAVED VERSIONS OF TeX PARAMETERS
1435 %
1436 % \normalbaselineskip and \normallineskip hold the
1437 % normal values of \baselineskip and \lineskip
1438
1439 % Any font-changing commands that change the normal value of \lineskip
1440 % and \baselineskip should change their saved values.
1441
1442 % The following definitions save token space. E.g., using \@height
1443 % instead of height saves 5 tokens at the cost in time of one macro
1444 % expansion.
1445
1446 \def\@height{height}
1447 \def\@depth{depth}
1448 \def\@width{width}
1449
1450 % The following implements the LaTeX \{ and \} commands.
1451 % Changed 21 Apr 87 to make them robust.
1452
1453 \def\{\{\protect\@lb}
1454 \def\@lb{\relax\ifmmode\lbrace\else$\m@th\lbrace$\fi}
1455 \def\}\{\protect\@rb}
1456 \def\@rb{\relax\ifmmode\rbrace\else$\m@th\rbrace$\fi}
1457
1458 \message{counters,}
1459 % *****
1460 % *          COUNTERS, ETC.          *
1461 % *****
1462 %
1463 % THE FOLLOWING ARE FROM PLAIN:
1464 % \z@      : A zero dimen or number. It's more efficient to write
1465 %           \parindent\z@ than \parindent Opt.
1466 % \@ne    : The number 1.
1467 % \m@ne   : The number -1.
1468 % \tw@    : The number 2.
1469 % \sixt@n : The number 16.
1470 % \@m     : The number 1000.
1471 % \@xxvi  : The number 32
1472 % \@M     : The number 10000.
1473 % \@Mi    : The number 10001.
1474 % \@Mii   : The number 10002.
1475 % \@Miii  : The number 10003.
1476 % \@Miv   : The number 10004.
1477 % \@MM    : The number 20000.
1478 %

```

```

1479 % \@flushglue : Glue used for \right- & \leftskip to = Opt plus 1fil
1480
1481 \chardef\@xxxii=32
1482 \mathchardef\@Mi=10001
1483 \mathchardef\@Mii=10002
1484 \mathchardef\@Miii=10003
1485 \mathchardef\@Miv=10004
1486
1487 % Redefine PLAIN.TEX macros not to be \outer
1488
1489 \def\newcount{\alloc@0\count\countdef\insc@unt}
1490 \def\newdimen{\alloc@1\dimen\dimendef\insc@unt}
1491 \def\newskip{\alloc@2\skip\skipdef\insc@unt}
1492 \def\newbox{\alloc@4\box\chardef\insc@unt}
1493 \def\newwrite{\alloc@7\write\chardef\sixt@n}
1494
1495 \newwrite\@unused
1496 \newcount\@tempcnta
1497 \newcount\@tempcntb
1498 \newif\if@tempswa\@tempswatrue
1499
1500 \newdimen\@tempdima
1501 \newdimen\@tempdimb
1502
1503 \newbox\@tempboxa
1504
1505 \newskip\@flushglue \@flushglue = Opt plus 1fil
1506 \newskip\@tempkipa
1507 \newskip\@tempkipb
1508 \newtoks\@temptokena
1509
1510 \message{hacks,}
1511 % *****
1512 % *          USEFUL HACKS          *
1513 % *****
1514 %
1515 % \@namedef{NAME} : Expands to \def\NAME , except name can contain any
1516 %                   characters.
1517 % \@nameuse{NAME} : Expands to \NAME .
1518 %
1519 % \@ifnextchar X{YES}{NO}
1520 %                   : Expands to YES if next character is an 'X',
1521 %                   and to NO otherwise. (Uses temps a-c.)
1522 %                   NOTE: GOBBLES ANY SPACE FOLLOWING IT.
1523 %
1524 % \@ifstar{YES}{NO} : Gobbles following spaces and then tests if next the
1525 %                   character is a '*'. If it is, then it gobbles the
1526 %                   '*' and expands to YES, otherwise it expands to NO.
1527 %
1528 % \@dblarg{CMD}{ARG} : \@dblarg{CMD}{ARG} expands to CMD[ARG]{ARG}. Use
1529 %                   \@dblarg\CS when \CS takes arguments [ARG1]{ARG2},
1530 %                   where default is ARG1 = ARG2.
1531 %
1532 % \@ifundefined{NAME}{YES}{NO}
1533 %                   : If \NAME is undefined then it executes YES,
1534 %                   otherwise it executes NO. More precisely,
1535 %                   true if \NAME either undefined or = \relax.
1536 % \@ifdefinable \NAME {YES}
1537 %                   : Executes YES if the user is allowed to define \NAME,
1538 %                   otherwise it gives an error. The user can define \NAME
1539 %                   if \@ifundefined{NAME} is true, 'NAME' /= 'relax'
1540 %                   and the first three letters of 'NAME' are not
1541 %                   'end'.
1542 % \newcommand{\FOO}[i]{TEXT}
1543 %                   : User command to define \FOO to be a macro with
1544 %                   i arguments (i = 0 if missing) having the definition
1545 %                   TEXT. Produces an error if \FOO already defined.
1546 %
1547 % \renewcommand{\FOO}[i]{TEXT} : Same as \newcommand, except it
1548 %                   checks if \FOO already defined.
1549 %
1550 % \newenvironment{FOO}[i]{DEF1}{DEF2}
1551 %                   equivalent to
1552 %                   \newcommand{\FOO}[i]{DEF1} \def{\endFOO}{DEF2}

```



```

1553 %
1554 % \renewenvironment : obvious companion to \newenvironment
1555 %
1556 % \@cons : See description of \output routine.
1557 %
1558 % \@car T1 T2 ... Tn\@nil == T1 (unexpanded)
1559 %
1560 % \@cdr T1 T2 ... Tn\@nil == T2 ... Tn (unexpanded)
1561 %
1562 % \typeout{message} : produces a warning message on the terminal
1563 %
1564 % \@warning{message}: prints 'LaTeX Warning: message.'
1565 %           With TeX 3.x, it also prints line number.
1566 %           (Changed 24 Jun 91 RmS)
1567 % \@@warning{message}: like \@warning, except that it never prints
1568 %           the line number (added 24 Jun 91 RmS).
1569 %
1570 % \typein{message} : Types message, asks the user to type in a command, then
1571 %           executes it
1572 %
1573 % \typein[CS]{MSG} : Same as above, except defines CS to be the input
1574 %           instead of executing it.
1575 %
1576 %% RmS 92/03/18: changed input channel from 0 to \@inputcheck to avoid conflicts
1577 %%           with other channels allocated by \newread
1578 \newread\@inputcheck
1579 \def\typein{\let\@typein\relax\@ifnextchar[{\@xtypein}{\@xtypein[\@typein]}}
1580 \def\@xtypein[#1]#2{\typeout{#2}\read\@inputcheck to#1\ifx #1\defpar \def#1{}\else
1581   \@iden{\expandafter\@strip\expandafter
1582     #1#1\@gobble\@gobble} \@gobble\fi\@typein}
1583 \def\@strip#1#2 \@gobble{\def #1{#2}}
1584 \def\@defpar{\par}
1585 \def\@iden#1{#1}
1586
1587 \ifx\inputlineno\undefined
1588   \let\on@line\empty
1589 \else
1590   \ifnum\inputlineno=\m@ne
1591     \let\on@line\empty
1592   \else
1593     \def\on@line{ on input line \the\inputlineno}
1594   \fi
1595 \fi
1596
1597 \def\typeout#1{\let\protect\string\immediate\write\@unused{#1}}
1598 \def\@@warning#1{\typeout{LaTeX Warning: #1.}}
1599 \def\@warning#1{\@@warning{#1\on@line}}
1600 \def\@namedef#1{\expandafter\def\csname #1\endcsname}
1601 \def\@nameuse#1{\csname #1\endcsname}
1602
1603 \def\@cons#1#2{\begingroup\let\@elt\relax\xdef#1{#1\@elt #2}\endgroup}
1604
1605 \def\@car#1#2\@nil{#1}
1606 \def\@cdr#1#2\@nil{#2}
1607
1608 % \@carcube T1 ... Tn\@nil = T1 T2 T3 , n > 3
1609 \def\@carcube#1#2#3#4\@nil{#1#2#3}
1610
1611 \def\newcommand#1{\@ifnextchar [{\@argdef#1}{\@argdef#1[0]}}
1612
1613 \def\renewcommand#1{\edef\@tempa{\expandafter\@cdr\string
1614   #1\@nil}\@ifundefined{\@tempa}{\@latexerr{\string#1\space undefined}\@ehc
1615   }{\@ifnextchar [{\@reargdef#1}{\@reargdef#1[0]}}
1616
1617 \def\newenvironment#1{\@ifnextchar
1618   [ {\@newenv{#1}}{\@newenv{#1}[0]}
1619
1620 \long\def\@newenv#1[#2]#3#4{\expandafter\newcommand
1621   \csname #1\endcsname[#2]{#3}\long
1622   \expandafter\def\csname end#1\endcsname{#4}}
1623
1624 \def\renewenvironment#1{\@ifnextchar
1625   [ {\@renewenv{#1}}{\@renewenv{#1}[0]}
1626

```

```

1627 \long\def\@renewenv#1[#2]#3#4{\expandafter\renewcommand
1628   \csname #1\endcsname[#2]{#3}\long
1629   \expandafter\def\csname end#1\endcsname{#4}}
1630
1631 \long\def\@argdef#1[#2]#3{\@ifdefinable #1{\@reargdef#1[#2]{#3}}}
1632
1633 % Absolutely untypable control sequence \@?? substituted for \@tempb in
1634 % definition of \@reargdef because it (and therefore \newcommand and
1635 % \renewcommand) leaves the control sequence dangerously \let to #.
1636 % (Change made 23 November 87.)
1637 %
1638 \catcode'\?=11\relax
1639 \long\def\@reargdef#1[#2]#3{\@tempcnta#2\relax\let#1\relax
1640 \edef\@tempa{\long\def#1}\@tempcntb \@ne
1641 \let\@???\relax\@whilenum\@tempcnta>\z@
1642 \do{\edef\@tempa{\@tempa\@???\the\@tempcntb}\advance\@tempcntb \@ne \advance
1643 \@tempcnta \m@ne}\let\@???\@tempa{#3}}
1644 \catcode'\?=12\relax
1645
1646
1647 % 9 Jan 90 : Missing % added to following definition.
1648 \long\def\@ifdefinable #1#2{\edef\@tempa{\expandafter\@cdr\string #1\@nil}%
1649 \@ifundefined{\@tempa}{\edef\@tempb{\expandafter\@carcube \@tempa xxxx\@nil}%
1650 \ifx \@tempb\@qend \@notdefinable}else
1651 \ifx \@tempa\@qrelax \@notdefinable}else #2\fi\fi}{\@notdefinable}}
1652
1653 \long\def\@ifundefined#1#2#3{\expandafter\ifx\csname
1654   #1\endcsname\relax#2}else#3\fi}
1655
1656
1657 % The following define \@qend and \@qrelax to be the strings 'end' and
1658 % 'relax' with the characters \catcoded 12.
1659
1660 \edef\@qend{\expandafter\@cdr\string@end\@nil}
1661 \edef\@qrelax{\expandafter\@cdr\string\relax\@nil}
1662
1663 % \@ifnextchar X{YES}{NO}
1664 % BEGIN
1665 %   \@tempe := X % uses \let
1666 %   \@tempa := YES
1667 %   \@tempb := NO
1668 %   \futurelet\@tempc
1669 %   \@ifnch
1670 % END
1671 %
1672 % \@ifnch ==
1673 % BEGIN
1674 %   if \@tempc = blank space
1675 %     then \@tempd := def(\@xifnch)
1676 %     else if \@tempc = \@tempe
1677 %       then \@tempd := def(\@tempa)
1678 %       else \@tempd := def(\@tempb)
1679 %     fi
1680 %   fi
1681 %   \@tempd
1682 % END
1683 %
1684 % \@ifnch ==
1685 % BEGIN
1686 %   gobble blanks
1687 %   \futurelet\@tempc
1688 %   \@ifnch
1689 % END
1690 %
1691 \def\@ifnextchar#1#2#3{\let\@tempe #1\def\@tempa{#2}\def\@tempb{#3}\futurelet
1692   \@tempc\@ifnch}
1693 \def\@ifnch{\ifx \@tempc \@sptoken \let\@tempd\@xifnch
1694   \else \ifx \@tempc \@tempe\let\@tempd\@tempa}else\let\@tempd\@tempb\fi
1695   \fi \@tempd}
1696
1697 % NOTE: the following hacking must precede the definition of \:
1698 % as math medium space.
1699
1700 \def\:{\let\@sptoken= } \: % this makes \@sptoken a space token

```

```

1701
1702 \def\:{\@xifnch} \expandafter\def\:{\futurelet\@tempc\@ifnch}
1703
1704 \def\@ifstar#1#2{\@ifnextchar *{\def\@tempa*{#1}\@tempa}{#2}}
1705
1706 \long\def\@dblarg#1{\@ifnextchar[{\#1}{\@xdblarg{#1}}}
1707 \long\def\@xdblarg#1#2#1[{\#2}]{#2}}
1708
1709 % The command \@sanitize changes the catcode of all special characters
1710 % except for braces to 'other'. It can be used for commands like
1711 % \index that want to write their arguments verbatim. Needless to
1712 % say, this command should only be executed within a group, or chaos
1713 % will ensue.
1714
1715 \def\@sanitize{\@makeother\ \@makeother\\\@makeother\$\@makeother\&%
1716 \@makeother\#\@makeother\^\@makeother\_ \@makeother\% \@makeother\~}
1717
1718
1719 \message{errors,}
1720 % *****
1721 % *          ERROR HANDLING          *
1722 % *****
1723 %
1724 % \@latexerr{MSG}{HLP}: Types a LaTeX error message MSG and gives an error
1725 % halt with error help message HLP.
1726 %
1727 \newlinechar`^^J
1728
1729 % 19 Jun 86, took out the grouping. re: John Hobby
1730 \def\@latexerr#1#2{%
1731 \edef\@tempc{#2}\errhelp\expandafter{\@tempc}%
1732 \typeout{LaTeX error. \space See LaTeX manual for explanation.^^J
1733 \space\@spaces\@spaces\@spaces Type \space H <return> \space for
1734 immediate help.}\errmessage{#1}}
1735
1736 \def\@spaces{\space\space\space\space}
1737
1738 %% error help message pieces.
1739 \def\@eha{Your command was ignored.
1740 ^^JType \space I <command> <return> \space to replace it
1741 with another command,^^Jor \space <return> \space to continue without it.}
1742 \def\@ehb{You've lost some text. \space \@ehc}
1743 \def\@ehc{Try typing \space <return>
1744 \space to proceed.^^JIf that doesn't work, type \space X <return> \space to
1745 quit.}
1746 \def\@ehd{You're in trouble here. \space\@ehc}
1747
1748 % Here are all the error message-generating commands of LaTeX.
1749 %
1750 % \@notdefinable : Error message generated in \@ifdefinable from calls
1751 % by \newcommand, \newlength, \newtheorem specifying an
1752 % already-defined command name.
1753 %
1754 % \@nolnerr : Generated by \newline and \\ when called in vertical mode.
1755 %
1756 % '\... undefined' : Generated in \renewcommand.
1757 %
1758 % \@nocnterr : Generated by \setcounter, \addtocounter or \newcounter
1759 % for undefined counter.
1760 %
1761 % \@ctrerr : Called when trying to print the value of a counter
1762 % numbered by letters that's greater than 26.
1763 %
1764 % 'Environment --- undefined' : Issued by \begin for undefined environment.
1765 %
1766 % \@badend : Called by \end that doesn't match its \begin.
1767 %
1768 % \@badmath : Called by \[, \], \{ or \} when used in wrong mode.
1769 %
1770 % \@toodeep : Called by a list environment nested more than six levels
1771 % deep, or an enumerate or itemize nested more than four
1772 % levels.
1773 %
1774 % \@badpoptabs : Called by \endtabbing when not enough \poptabs have

```

```

1775 %           occurred, or by \poptabs when too many have occurred.
1776 %
1777 % \@badtab : Called by |>, |+ , |- or |< when stepping to an undefined tab.
1778 %
1779 % 'tab overflow' : Occurs in |= when maximum number of tabs exceeded.
1780 %
1781 % '|< in mid line' : Occurs in |< when it appears in middle of line.
1782 %
1783 % \@preamerr : Occurs in array or tabular environment, or in \multicolumn
1784 %           command, when error in argument detected.
1785 %
1786 % \@badlinearg : Occurs in \line and \vector command when a bad slope
1787 %           argument is encountered.
1788 %
1789 % \@parmoderr : Occurs in a float environment or a \marginpar when
1790 %           encountered in inner vertical mode.
1791 %
1792 % \@fltovf   : Occurs in float environment or \marginpar when there
1793 %           are no more free boxes for storing floats.
1794 %
1795 % \@latexbug  : Occurs in output routine. This is bad news.
1796 %
1797 % 'Float(s) lost' : In output routine, caused by a float environment or
1798 %           \marginpar occurring in inner vertical mode.
1799 %
1800 % \@nofonterror : Typeface not available. %% OBSOLETE; DELETED.
1801 %
1802 % \@badcrerr   : A \\ used where it shouldn't be in a centering or flushing
1803 %           environment.
1804 %
1805 % \@noitemerr  : \addvspace or \addpenalty was called when not in vmode.
1806 %           Probably caused by a missing \item.
1807 %
1808 % \@notprerr   : A command that can be used only in the preamble
1809 %           appears after the \begin{document} command.
1810 %
1811 \def\@notdefinable{\@latexerr{Command name '@tempa' already used}\@eha}
1812 %
1813 \def\@nolnerr{\@latexerr{There's no line here to end}\@eha}
1814 %
1815 \def\@nocnterr{\@latexerr{No such counter}\@eha}
1816 %
1817 \def\@ctrerr{\@latexerr{Counter too large}\@ehb}
1818 %
1819 \def\@nodocument{\@latexerr{Missing \string\begin{document}}\@ehd}
1820 %
1821 \def\@badend#1{\@latexerr{\string\begin{\@currenvir} ended by
1822 \string\end{#1}}\@eha}
1823 %
1824 \def\@badmath{\@latexerr{Bad math environment delimiter}\@eha}
1825 %
1826 \def\@toodeep{\@latexerr{Too deeply nested}\@ehd}
1827 %
1828 \def\@badpoptabs{\@latexerr{\string\pushtabs \space and \string\poptabs
1829 \space don't match}\@ehd}
1830 %
1831 \def\@badtab{\@latexerr{Undefined tab position}\@ehd}
1832 %
1833 \def\@preamerr#1{\@latexerr{\ifcase #1 Illegal character\or
1834 Missing @-exp\or Missing p-arg\fi\space
1835 in array arg}\@ehd}
1836 %
1837 \def\@badlinearg{\@latexerr{Bad \string\line\space or \string\vector
1838 \space argument}\@ehb}
1839 %
1840 \def\@parmoderr{\@latexerr{Not in outer par mode}\@ehb}
1841 %
1842 \def\@fltovf{\@latexerr{Too many unprocessed floats}\@ehb}
1843 %
1844 \def\@latexbug{\@latexerr{This may be a LaTeX bug}{Call for help}}
1845 %
1846 % \def\@nofonterror{\@latexerr{Typeface not available}\@eha}
1847 %
1848 \def\@badcrerr {\@latexerr{Bad use of \string\\}\@ehc}

```

```

1849
1850 \def\@noitemerr{\@latexerr{Something's wrong--perhaps a missing
1851 \string\item}\@ehc}
1852
1853 \def\@notprerr {\@latexerr{Can be used only in preamble}\@eha}
1854
1855 \message{par,}
1856 % *****
1857 % * \par AND \everypar *
1858 % *****
1859 %
1860 % There are two situations in which \par may be changed:
1861 %
1862 % - Long-term changes, in which the new value is to remain in effect
1863 % until the current environment is left. The environments that
1864 % change \par in this way are the following:
1865 %
1866 % * All list environments (itemize, quote, etc.)
1867 % * Environments that turn \par into a noop:
1868 % tabbing, array and tabular.
1869 %
1870 % - Temporary changes, in which \par is restored to its previous value the
1871 % next time it is executed. The following are all such uses.
1872 % * \end [when preceded by \@endparenv, which is called by
1873 % \endtrivlist]
1874 % * The mechanism for avoiding page breaks and getting the
1875 % spacing right after section heads.
1876 %
1877 % To permit the proper interaction of these two situations, long-term
1878 % changes are made by the following command:
1879 % \setpar{VAL} : To set \par. It \def's \par and \@par to VAL.
1880 % Short-term changes are made by the usual \def\par commands.
1881 % The original values are restored after a short-term change
1882 % by the \@restorepar commands.
1883 %
1884 % NOTE: \@par always is defined to be the original TeX \par.
1885 %
1886 % \everypar is changed only for the short term. Whenever \everypar
1887 % is set non-null, it should restore itself to null when executed.
1888 % The following commands change \everypar in this way:
1889 % * \item
1890 % * \end [when preceded by \@endparenv, which is called by
1891 % \endtrivlist]
1892 % * \minipage
1893 %
1894 % WARNING: Commands that make short-term changes to \par and \everypar
1895 % must take account of the possibility that the new commands and the
1896 % ones that do the restoration may be executed inside a group. In
1897 % particular, \everypar is executed inside a group whenever a new paragraph
1898 % begins with a left brace. The \everypar command that restores its
1899 % definition should be local to the current group (in case the command
1900 % is inside a minipage used inside someplace where \everypar has been
1901 % redefined). Thus, if \everypar is redefined to do an \everypar{}
1902 % it could take several executions of \everypar before
1903 % the restoration 'holds'. This usually causes no problem. However, to
1904 % prevent the extra executions from doing harm, use a global switch
1905 % to keep anything harmful in the new \everypar from being done twice.
1906 %
1907 % WARNING: Commands that change \everypar should remember that \everypar
1908 % might be supposed to set the following switches false:
1909 % @nobreak
1910 % @minipage
1911 % they should do the setting if necessary.
1912
1913 \def\@par{\let\par\@par\par}
1914
1915 \def\@setpar#1{\def\par{#1}\def\@par{#1}}
1916 \def\@restorepar{\def\par{\@par}}
1917
1918 \message{spacing,}
1919 % *****
1920 % * SPACING / LINE AND PAGE BREAKING *
1921 % *****
1922 %

```

```

1923 % USER COMMANDS:
1924 % \nopagebreak[i] : i = 0,...,4. Default argument = 4. Puts a penalty
1925 %           into the vertical list output as follows:
1926 %           0 : penalty = 0
1927 %           1 : penalty = \@lowpenalty
1928 %           2 : penalty = \@medpenalty
1929 %           3 : penalty = \@highpenalty
1930 %           4 : penalty = 10000
1931 % \pagebreak[i] : same as \nopagebreak except negatives of its penalty
1932 % \linebreak[i], \nolinebreak[i] : analogs of the above
1933 % \samepage : inhibits page breaking most places by setting the following
1934 %           penalties to 10000
1935 %           \interlinepenalty
1936 %           \postdisplaypenalty
1937 %           \interdisplaylinepenalty
1938 %           \@beginparpenalty
1939 %           \@endparpenalty
1940 %           \@itempenalty
1941 %           \@secpenalty
1942 %           \interfootnotelinepenalty
1943 %
1944 % \obeycr : defines <CR> == \\.
1945 % \restorecr : restores <CR> to its usual meaning.
1946 %
1947 % \ : initially defined to be \newline
1948 % \[LENGTH] : initially defined to be \vspace{LENGTH}\newline
1949 %           Note: \* adds a \adjust{penalty 10000}
1950 %
1951 \def\nopagebreak{\@ifnextchar[{\@nopgbk}{\@nopgbk[4]}}
1952 \def\@nopgbk[#1]{\ifvmode \penalty \@getpen{#1}\else
1953 \@bsphack\adjust{\penalty \@getpen{#1}}\@esphack\fi}
1954 %
1955 \def\pagebreak{\@ifnextchar[{\@pgbk}{\@pgbk[4]}}
1956 \def\@pgbk[#1]{\ifvmode \penalty -\@getpen{#1}\else
1957 \@bsphack\adjust{\penalty -\@getpen{#1}}\@esphack\fi}
1958 %
1959 \def\nolinebreak{\@ifnextchar[{\@nolnkb}{\@nolnkb[4]}}
1960 \def\@nolnkb[#1]{\ifvmode \nolnerr\else \@tempkipa\lastskip
1961 \unskip \penalty \@getpen{#1}\ifdim \@tempkipa >\z@
1962 \hskip\@tempkipa\ignorespaces\fi\fi}
1963 %
1964 \def\linebreak{\@ifnextchar[{\@lnbk}{\@lnbk[4]}}
1965 \def\@lnbk[#1]{\ifvmode \nolnerr\else
1966 \unskip\penalty -\@getpen{#1}\fi}
1967 %
1968 \def\samepage{\interlinepenalty\@M
1969 \postdisplaypenalty\@M
1970 \interdisplaylinepenalty\@M
1971 \@beginparpenalty\@M
1972 \@endparpenalty\@M
1973 \@itempenalty\@M
1974 \@secpenalty\@M
1975 \interfootnotelinepenalty\@M}
1976 %
1977 % \nobreak added to \newline to prevent null lines when \newline
1978 % ends an overfull line. Change made 24 May 89 as suggested by
1979 % Frank Mittelbach and Rainer Sch\"opf
1980 %
1981 \def\newline{\ifvmode \@nolnerr \else \unskip\nobreak\hfil
1982 \penalty -\@M\fi}
1983 %
1984 %
1985 \def\@normalcr{\@ifstar{\adjust{\penalty\@M}}{\@xnewline}}{\@xnewline}}
1986 %
1987 \def\@xnewline{\@ifnextchar[{\@newline}}{\@newline}}
1988 %
1989 \def\@newline[#1]{\ifhmode\unskip\fi\vspace{#1}\newline}
1990 %
1991 \let\@=\@normalcr
1992 %
1993 \def\@getpen#1{\ifcase #1 0 \or \@lowpenalty\or
1994 \@medpenalty \or \@highpenalty
1995 \else \@M \fi}
1996 %

```

```

1997 % @nobreak : Switch used to avoid page breaks caused by \label after a section
1998 %           heading, etc. It should be GLOBALLY set true after the \nobreak
1999 %           and GLOBALLY set false by the next invocation of \everypar.
2000 %           Commands that reset \everypar should globally set it false
2001 %           if appropriate.
2002 %
2003 \newif\if@nobreak \@nobreakfalse
2004
2005 % \@bsphack ... \@esphack
2006 %   used by macros such as \index and \begin{@float} ... \end{@float}
2007 %   that want to be invisible -- i.e.,
2008 %   not leave any extra space when used in the middle of text. Such
2009 %   a macro should begin with \@bsphack and end with \@esphack
2010 %   The macro in question should not create any text, nor change the
2011 %   mode.
2012 %
2013 % \@Esphack is a variant of \@esphack that sets the @ignore switch to true
2014 %   (as \@esphack used to do previously). This is currently used only
2015 %   for float and similar environments.
2016 %
2017 % \@bsphack ==
2018 % BEGIN
2019 %   if not mmode then           %% Test for math mode added 18 Dec 89
2020 %       \dimen@savsk := \lastskip
2021 %       if hmode then \@savsf := \spacefactor fi
2022 %   fi
2023 % END
2024 %
2025 % \@esphack ==
2026 % BEGIN
2027 %   if not mmode then           %% Test for math mode added 18 Dec 89
2028 %       if hmode
2029 %           then \spacefactor := \@savsf
2030 %               if \dimen@savsk > Opt then \ignorespaces fi
2031 %       fi
2032 %   fi
2033 % END
2034 %
2035 % \@Esphack ==
2036 % BEGIN
2037 %   if not mmode then
2038 %       if hmode
2039 %           then \spacefactor := \@savsf
2040 %               if \dimen@savsk > Opt then \ignorespaces
2041 %                   \global\@ignoretrue fi
2042 %       fi
2043 %   fi
2044 % END
2045 %
2046
2047 \newdimen@savsk
2048 \newcount@savsf
2049
2050 \def\@bsphack{\relax\ifmmode\else\@savsk\lastskip
2051   \ifhmode\@savsf\spacefactor\fi\fi}
2052
2053 \def\@esphack{\relax\ifmmode\else\ifhmode\spacefactor\@savsf
2054   \ifdim \@savsk >\z@ \ignorespaces
2055   \fi \fi\fi}
2056
2057 \def\@Esphack{\relax\ifmmode\else\ifhmode\spacefactor\@savsf
2058   \ifdim \@savsk >\z@ \global\@ignoretrue \ignorespaces
2059   \fi \fi\fi}
2060
2061 % VERTICAL SPACING:
2062 %
2063 % LaTeX supports the PLAIN TeX commands \smallskip, \medskip and \bigskip.
2064 % However, it redefines them using \vspace instead of \vskip.
2065 %
2066 % Extra vertical space is added by the command command \addvspace{SKIP},
2067 % which adds a vertical skip of SKIP to the document. The sequence
2068 % \addvspace{S1} \addvspace{S2}
2069 % is equivalent to
2070 % \addvspace{maximum of S1, S2}.

```

```

2071 % \addvspace should be used only in vertical mode, and gives an error if it's
2072 % not. The \addvspace command does NOT add vertical space if
2073 % @minipage = T. The minipage environment uses this to inhibit
2074 % the addition of extra vertical space at the beginning.
2075 %
2076 % Penalties are put into the vertical list with the \addpenalty{PENALTY}
2077 % command. It works properly when \addpenalty and \addvspace commands
2078 % are mixed.
2079 %
2080 % The @nobreak switch is set true used when in vertical mode and no page
2081 % break should occur. (Right now, it is used only by the section heading
2082 % commands to inhibit page breaking after a heading.)
2083 %
2084 %
2085 % \addvspace{SKIP} ==
2086 % BEGIN
2087 %   if vmode
2088 %     then if @minipage
2089 %       else if \lastskip = 0
2090 %         then \vskip SKIP
2091 %       else if \lastskip < SKIP
2092 %         then \vskip -\lastskip
2093 %           \vskip SKIP
2094 %       else if SKIP < 0 and \lastskip >= 0
2095 %         then \vskip -\lastskip
2096 %           \vskip \lastskip + SKIP
2097 %     fi
2098 %   else 'missing \item' error.
2099 % fi
2100 % END
2101
2102 \def\addvspace#1{\ifvmode
2103   \if@minipage\else
2104     \ifdim \lastskip =\z@ \vskip #1\relax
2105     \else \@tempskipb#1\relax\@xaddvskip
2106   \fi\fi
2107   \else\@noitemerr\fi}
2108
2109 \def\@xaddvskip{\ifdim \lastskip <\@tempskipb\vskip -\lastskip\vskip
2110   \@tempskipb
2111   \else \ifdim \@tempskipb<\z@
2112     \ifdim \lastskip <\z@
2113       \else \advance\@tempskipb\lastskip
2114       \vskip -\lastskip \vskip \@tempskipb
2115   \fi\fi\fi}
2116
2117 \def\addpenalty#1{\ifvmode
2118   \if@minipage\else\if@nobreak\else
2119     \ifdim\lastskip=\z@ \penalty#1\relax
2120     \else \@tempskipb\lastskip
2121     \vskip -\lastskip \penalty#1\vskip \@tempskipb
2122   \fi\fi\fi
2123   \else\@noitemerr\fi}
2124
2125 \def\vspace{\@ifstar{\@vspacer}{\@vspace}}
2126 \def\@vspace#1{\ifvmode
2127   \dimen@\prevdepth \vskip #1\vskip\z@ \prevdepth\dimen@
2128   \else
2129     \@bsphack\vadjust{\dimen@\prevdepth
2130       \vskip #1\vskip\z@ \prevdepth\dimen@}\@esphack\fi}
2131 \def\@vspacer#1{\ifvmode \dimen@\prevdepth
2132   \hrule \@height\z@ \nobreak \vskip #1\vskip\z@
2133   \prevdepth\dimen@
2134   \else
2135     \@bsphack\vadjust{\dimen@\prevdepth \hrule \@height\z@ \nobreak
2136     \vskip #1\vskip\z@ \prevdepth\dimen@}\@esphack\fi}
2137
2138 \def\smallskip{\vspace\smallskipamount}
2139 \def\medskip{\vspace\medskipamount}
2140 \def\bigskip{\vspace\bigskipamount}
2141
2142
2143 % See list environment for explanation of the following macros.
2144

```



```

2145 \def\endtrivlist{\if@newlist\@noitemerr\fi
2146   \if@inlabel\indent\fi
2147   \ifhmode\unskip \par\fi
2148   \if@nparlist \else
2149     \ifdim\lastskip >\z@ \@tempskipa\lastskip \vskip -\lastskip
2150     \advance\@tempskipa\parskip \advance\@tempskipa -\@outerparskip
2151     \vskip\@tempskipa
2152   \fi\@endparenv\fi}
2153
2154 % CHANGES TO \@endparenv:
2155 % Changed \hskip -\parindent to \setbox0=\lastbox so a \noindent
2156 % becomes a no-op when used before a line immediately following a
2157 % list environment. (Changed 23 Oct 86)
2158 %
2159 % To suppress the paragraph indentation in text immediately following
2160 % a paragraph-making environment, \everypar is changed to remove the
2161 % space, and \par is redefined to restore \everypar. Instead of redefining
2162 % \par and \everypar, \@endparenv was changed to set the @endpe switch,
2163 % letting \end redefine \par and \everypar. This allows paragraph-
2164 % making environments work right when called by other environments.
2165 % (Changed 27 Oct 86)
2166
2167 \def\@endparenv{\addpenalty\@endparpenalty\addvspace\@topsepadd\@endpetrue}
2168
2169 \def\@doendpe{\@endpetrue
2170   \def\par{\@restorepar\everypar{}\par\@endpefalse}\everypar
2171   {\setbox\z@\lastbox\everypar{}\@endpefalse}}
2172
2173 \newif\if@endpe
2174 \@endpefalse
2175
2176 % HORIZONTAL SPACE
2177 %
2178 % \, : used in paragraph mode produces a \thinspace. It has the ordinary
2179 % definition in math mode. Useful for quotes inside quotes, as in
2180 % ‘\, ‘Foo’, he said.’
2181 %
2182 % \@ : placed before a ‘.’, makes it a sentence-ending period. Does the
2183 % right thing for other punctuation marks as well. Does this by
2184 % setting spacefactor to 1000.
2185
2186 \def\,\{\protect\pcomma}
2187 \def\pcomma{\relax\ifhmode\mskip\thinmuskip\else\thinspace\fi}
2188
2189
2190 \def\@{\spacefactor\@m}
2191
2192 \def\hspace{\protect\phspace}
2193 \def\phspace{\@ifstar{\@hspacer}{\@hspace}}
2194 \def\@hspace#1{\leavevmode\hskip #1\relax}
2195
2196 \def\@hspacer#1{\leavevmode\vrule \@width\z@\nobreak
2197   \hskip #1\hskip \z@skip}
2198   %% extra \hskip Opt added 12/17/85 to guard
2199   %% against a following \unskip
2200   %% \relax added 13 Oct 88 for usual TeX lossage
2201   %% replaced both changes by \hskip\z@skip 27 Nov 91
2202
2203 % define \fill to = Opt plus 1fill
2204 \newskip\fill \fill = Opt plus 1fill
2205
2206 % \stretch{N} == Opt plus N fill
2207 \def\stretch#1{\z@ plus #1fill\relax}
2208
2209 {\catcode'\^M=13 \gdef\obeycr{\catcode'\^M=13 \def^^M{\}\@gobblecr}%
2210 \gdef\restorecr{\catcode'\^M=5 }} %} BRACE MATCHING
2211
2212
2213 \message{control,}
2214 % *****
2215 % * PROGRAM CONTROL STRUCTURE MACROS *
2216 % *****
2217 %
2218 % \@whilenum TEST \do {BODY}

```

```

2219 % \@whiledim TEST \do {BODY} : These implement the loop
2220 %   while TEST do BODY od
2221 %   where TEST is a TeX \ifnum or \ifdim test, respectively.
2222 %   They are optimized for the normal case of TEST initially false.
2223 %
2224 % \@whilesw SWITCH \fi {BODY} : Implements the loop
2225 %   while SWITCH do BODY od
2226 %   where SWITCH is a command defined by \news witch.
2227 %   Optimized for normal case of SWITCH initially false.
2228 %
2229 % \@for NAME := LIST \do {BODY} : Assumes that LIST expands to A1,A2, ... ,An .
2230 %   Executes BODY n times, with NAME = Ai on the i-th iteration.
2231 %   Optimized for the normal case of n = 1. Works for n=0.
2232 %
2233 % \@tfor NAME := LIST \do {BODY}
2234 %   if, before expansion, LIST = T1 ... Tn where each Ti is a
2235 %   token or {...}, then executes BODY n times, with NAME = Ti
2236 %   on the i-th iteration. Works for n=0.
2237 %
2238 % NOTES: 1. These macros use no \@temp sequences.
2239 %         2. These macros do not work if the body contains anything that looks
2240 %         syntactically to TeX like an improperly balanced \if \else \fi.
2241 %
2242 % \@whilenum TEST \do {BODY} ==
2243 % BEGIN
2244 %   if TEST
2245 %     then BODY
2246 %       \@whilenum{TEST \relax BODY}
2247 % END
2248 %
2249 % \@iwhilenum {TEST BODY} ==
2250 % BEGIN
2251 %   if TEST
2252 %     then BODY
2253 %       \@nextwhile = def(\@iwhilenum)
2254 %     else \@nextwhile = def(\@whilenoop)
2255 %   fi
2256 %   \@nextwhile {TEST BODY}
2257 % END
2258 %
2259 % \@whilesw SWITCH \fi {BODY} ==
2260 % BEGIN
2261 %   if SWITCH
2262 %     then BODY
2263 %       \@iwhilesw {SWITCH BODY}\fi
2264 %   fi
2265 % END
2266 %
2267 % \@iwhilesw {SWITCH BODY} \fi ==
2268 % BEGIN
2269 %   if SWITCH
2270 %     then BODY
2271 %       \@nextwhile = def(\@iwhilesw)
2272 %     else \@nextwhile = def(\@whilesnoop)
2273 %   fi
2274 %   \@nextwhile {SWITCH BODY} \fi
2275 % END
2276 %
2277 \def\@whilenoop#1{}
2278 \def\@whilenum#1\do #2{\ifnum #1\relax #2\relax\@iwhilenum{#1\relax
2279   #2\relax}\fi}
2280 \def\@iwhilenum#1{\ifnum #1\let\@nextwhile\@iwhilenum
2281   \else\let\@nextwhile\@whilenoop\fi\@nextwhile{#1}}
2282 %
2283 \def\@whiledim#1\do #2{\ifdim #1\relax#2\@iwhiledim{#1\relax#2}\fi}
2284 \def\@iwhiledim#1{\ifdim #1\let\@nextwhile\@iwhiledim
2285   \else\let\@nextwhile\@whilenoop\fi\@nextwhile{#1}}
2286 %
2287 \long\def\@whilesnoop#1\fi{}
2288 \long\def\@whilesw#1\fi#2{#1#2\@iwhilesw{#1#2}\fi\fi}
2289 \long\def\@iwhilesw#1\fi{#1\let\@nextwhile\@iwhilesw
2290   \else\let\@nextwhile\@whilesnoop\fi\@nextwhile{#1}\fi}
2291 %
2292 % \@for NAME := LIST \do {BODY} ==

```

```

2293 % BEGIN \@forloop expand(LIST),\@nil,\@nil \@@ NAME {BODY} END
2294 %
2295 % \@forloop CAR, CARCDR, CDRCDR \@@ NAME {BODY} ==
2296 % BEGIN
2297 % NAME = CAR
2298 % if def(NAME) = def(\@nnil)
2299 % else BODY;
2300 % NAME = CARCDR
2301 % if def(NAME) = def(\@nnil)
2302 % else BODY
2303 % \@iforloop CDRCDR \@@ NAME \do {BODY}
2304 % fi
2305 % fi
2306 % END
2307 %
2308 % \@iforloop CAR, CDR \@@ NAME {BODY} =
2309 % NAME = CAR
2310 % if def(NAME) = def(\@nnil)
2311 % then \@nextwhile = def(\@fornoop)
2312 % else BODY ;
2313 % \@nextwhile = def(\@iforloop)
2314 % fi
2315 % \@nextwhile name cdr {body}
2316 %
2317 % \@tfor NAME := LIST \do {BODY}
2318 % = \@tforloop LIST \@nil \@@ NAME {BODY}
2319 %
2320 % \@tforloop car cdr \@@ name {body} =
2321 % name = car
2322 % if def(name) = def(\@nnil)
2323 % then \@nextwhile == \@fornoop
2324 % else body ;
2325 % \@nextwhile == \@forloop
2326 % fi
2327 % \@nextwhile name cdr {body}
2328 %
2329
2330 \def\@nnil{\@nil}
2331 \def\@empty{}
2332 \def\@fornoop#1\@#2#3{}
2333
2334 \def\@for#1:=#2\do#3{\edef\@fortmp{#2}\ifx\@fortmp\@empty \else
2335 \expandafter\@forloop#2,\@nil,\@nil\@#1{#3}\fi}
2336
2337 \def\@forloop#1,#2,#3\@#4#5{\def#4{#1}\ifx #4\@nnil \else
2338 #5\def#4{#2}\ifx #4\@nnil \else#5\@iforloop #3\@#4{#5}\fi\fi}
2339
2340 \def\@iforloop#1,#2\@#3#4{\def#3{#1}\ifx #3\@nnil
2341 \let\@nextwhile\@fornoop \else
2342 #4relax\let\@nextwhile\@iforloop\fi\@nextwhile#2\@#3{#4}}
2343
2344 %%RmS 91/10/17: Corrected bug in \@tfor: \xdef replaced by \def
2345 %% (See FMi's array.doc)
2346 \def\@tfor#1:=#2\do#3{\def\@fortmp{#2}\ifx\@fortmp\@empty \else
2347 \@tforloop#2\@nil\@nil\@#1{#3}\fi}
2348 \def\@tforloop#1#2\@#3#4{\def#3{#1}\ifx #3\@nnil
2349 \let\@nextwhile\@fornoop \else
2350 #4relax\let\@nextwhile\@tforloop\fi\@nextwhile#2\@#3{#4}}
2351
2352
2353 \message{files,}
2354 %
2355 % ***** FILE HANDLING *****
2356 %
2357 %
2358 % THE FOLLOWING USER COMMANDS ARE DEFINED IN THIS PART:
2359 % \document : Reads in the .AUX files and \catcode's @ to 12.
2360 % \nofiles : Suppresses all file output by setting \@filesw false.
2361 % \includeonly{NAME1, ... ,NAMEn}
2362 % : Causes only parts NAME1, ... ,NAMEn to be read by
2363 % their \include commands. Works by setting \@partsw true
2364 % and setting \@partlist to NAME1, ... ,NAMEn.
2365 % \include{NAME} : Does an \input NAME unless \partsw is true and
2366 % NAME is not in \@partlist. If \@filesw is true, then

```

```

2367 %           it directs .AUX output to NAME.AUX, including a
2368 %           checkpoint at the end.
2369 % \input{NAME} : The same as TeX's \input, except it allows optional
2370 %           braces around the file name.
2371 %
2372 % VARIABLES, SWITCHES AND INTERNAL COMMANDS:
2373 % \@mainaux : Output file number for main .AUX file.
2374 % \@partaux : Output file number for current part's .AUX file.
2375 % \@auxout : Either \@mainout or \@partout, depending on which .AUX
2376 %           file output goes to.
2377 % \@input{foo} : If file foo exists, then \input's it, otherwise types
2378 %           a warning message.
2379 % @filesw : Switch -- set false if no .AUX, .TOC, .IDX etc files are
2380 %           to be
2381 % @partsw : Set true by a \includeonly command.
2382 % \@partlist : Set to the argument of the \includeonly command.
2383 %
2384 % \cp@FOO : The checkpoint for \include'd file FOO.TEX, written
2385 %           by \writeckpt at the end of file FOO.AUX
2386 %
2387 % \document ==
2388 % BEGIN
2389 % \endgroup % cancels \begingroup generated by \begin command
2390 % \@colht := \@colroom := \vsize := \textheight
2391 % \columnwidth := \textwidth
2392 % \@clubpenalty := \clubpenalty % \@clubpenalty saves value.
2393 % IF @twocolumn = T
2394 % THEN \columnwidth := (\columnwidth - \columnsep)/2
2395 % @firstcolumn := T
2396 % FI
2397 % \hsize := \linewidth := \columnwidth
2398 % \begingroup
2399 % \floatplacement \@dblfloatplacement
2400 % \input{\jobname.aux}
2401 % \endgroup
2402 % IF @filesw = T
2403 % THEN open file \@mainaux for writing
2404 % write '\relax' on file \@mainaux
2405 % FI
2406 % \do{COMMAND} == BEGIN \let COMMAND = \@notprerr END
2407 % \@preamblecmds
2408 % \do == \noexpand
2409 % \normalsize
2410 % \everypar{}
2411 % @noskipsec := F
2412 % END
2413 %
2414 % \includeonly{FILELIST} ==
2415 % BEGIN
2416 % \@partsw := T
2417 % \@partlist := FILELIST
2418 % END
2419 %
2420 % \include{FILE} ==
2421 % BEGIN
2422 % \clearpage
2423 % if @filesw = T
2424 % then \immediate\write\@mainaux{\string\@input{FILE.AUX}}
2425 % fi
2426 % if @partsw = T
2427 % then \@tempswa := F
2428 % \tempb == FILE
2429 % for \@tempa := \@partlist
2430 % do if eval(\@tempa) = eval(\@tempb)
2431 % then \@tempswa := T fi
2432 % od
2433 % fi
2434 %
2435 % if \@tempswa = T
2436 % then \@auxout := \@partaux
2437 % if @filesw = T
2438 % then \immediate\openout\@partaux{FILE.AUX}
2439 % \immediate\write\@partaux{\relax}
2440 % fi

```

```

2441 %           \@input{FILE.TEX}
2442 %           \clearpage
2443 %           \@writeckpt{FILE}
2444 %           if @filesw then \closeout \@partaux fi
2445 %           \@auxout := \@mainaux
2446 %           else \cp@FILE
2447 %           fi
2448 %           END
2449 %
2450 % \@writeckpt{FILE} ==
2451 % BEGIN
2452 %   if \@filesw = T
2453 %     \immediate\write on file \@partaux:
2454 %       \gdef\cp@FILE{           %% }
2455 %     for \@tempa := \cl@ckpt
2456 %       do \immediate\write on file \@partaux:
2457 %         \global\string\setcounter
2458 %           {eval(\@tempa)}{eval(\c@eval(\@tempa))}
2459 %       od
2460 %     \immediate\write on file \@partaux: }
2461 %   fi
2462 %   END
2463 %
2464 %   INITIALIZATION
2465 %   \@tempswa := T
2466 %
2467 \newif\if@filesw \@fileswtrue
2468 \newif\if@partsw \@partswfalse
2469 \newwrite\@mainaux
2470 \newwrite\@partaux
2471 %
2472 \newcount\@clubpenalty
2473 %
2474 %% FMi & RmS 91/08/26 set @noskipsec switch to true in the preamble
2475 %% and to false by \begin{document} to catch lists in the preamble,
2476 %% i.e., to produce a ‘‘nodocument’’ error when things like
2477 %% \maketitle appear before \begin{document}.
2478 %
2479 % \@noskipsectrue %% set below where switch is defined
2480 %
2481 % 91/03/26 FMi: |\process@table| added to support NFSS.
2482 % This will also work with old lfonts if no other style defines
2483 % |\process@table|.
2484 %
2485 \def\document{\endgroup
2486   \@colht\textheight \@colroom\textheight \vsize\textheight
2487   \columnwidth\textwidth \@clubpenalty\clubpenalty
2488   \if@twocolumn \advance\columnwidth -\columnsep
2489   \divide\columnwidth\tw@ \hsize\columnwidth \@firstcolumntrue
2490   \fi
2491   \hsize\columnwidth \linewidth\hsize
2492   \begingroup\@floatplacement\@dblfloatplacement
2493   \makeatletter\let\@writefile\@gobbletwo
2494   \@input{\jobname.aux}\endgroup
2495   \if@filesw \immediate\openout\@mainaux=\jobname.aux
2496   \immediate\write\@mainaux{\relax}\fi
2497   \csname process@table\endcsname
2498   \let\glb@currsiz@empty %% Force \baselineskip initialisation.
2499   \def\do##1{\let ##1\@notprerr}%
2500   \@preamblecmds
2501   \let\do\noexpand
2502   \@normalsize\everypar{}\@noskipsecfalse}
2503 %
2504 \def\@gobbletwo#1#2{}
2505 %
2506 \def\nofiles{\@fileswfalse \typeout
2507   {No auxiliary output files.}\typeout{}}
2508 %
2509 %% RmS 92/03/18: changed input channel from 1 to \@inputcheck to avoid
2510 %%           conflicts with other channels allocated by \newread
2511 \def\@input#1{\openin\@inputcheck #1 \ifeof\@inputcheck \typeout
2512   {No file #1.}\else\closein\@inputcheck \relax\@input #1 \fi}
2513 \let\@auxout=\@mainaux
2514 %

```

```

2515 \def\includeonly#1{\@partswtrue\edef\@partlist{#1}}
2516
2517 % In the definition of \include, \def\@tempb changed to \edef\@tempb to
2518 % be consistent with the \edef in \includeonly. (Suggested by Rainer
2519 % Sch\"opf & Frank Mittelbach. Change made 20 Jul 88.)
2520 %
2521 % Changed definition of \include to allow space at end of file name--
2522 % otherwise, typing \include{foo } would cause LaTeX to overwrite
2523 % foo.tex. Change made 24 May 89, suggested by Rainer Sch\"opf and
2524 % Frank Mittelbach
2525
2526 \def\include#1{\@include#1 }
2527 \def\@include#1 {\clearpage
2528 \if@filesw \immediate\write\@mainaux{\string\@input{#1.aux}}\fi
2529 \@tempswtrue\if@partsw \@tempswafalse\edef\@tempb{#1}\@for
2530 \@tempa:=\@partlist\do{\ifx\@tempa\@tempb\@tempswtrue\fi}\fi
2531 \if@tempswa \if@filesw \let\@auxout\@partaux
2532 \immediate\openout\@partaux #1.aux
2533 \immediate\write\@partaux{\relax}\fi\input{#1.tex}\clearpage
2534 \writetocpt{#1}\if@filesw \immediate\closeout\@partaux \fi
2535 \let\@auxout\@mainaux\else\@nameuse{cp@#1}\fi}
2536
2537 \def\@writetocpt#1{\if@filesw
2538 \immediate\write\@partaux{\string\global\string\@namedef{cp@#1}\@charlb}%
2539 {\let\@elt\@wckptelt \cl@ckpt}\immediate\write\@partaux{\@charrb}\fi}
2540
2541 \def\@wckptelt#1{\immediate\write\@partaux
2542 {\string\setcounter{#1}{\the\@nameuse{c@#1}}}}
2543
2544 \def\input{\@ifnextchar \bgroup{\@input}{\@input }}
2545 \def\@input#1{\@input #1 }
2546
2547 % The following defines \@charlb and \@charrb to be { and }, respectively
2548 % with \catcode 11.
2549 {\catcode' [=1 \catcode' ]=2
2550 \catcode' {=11 \catcode' }=11
2551 \gdef\@charlb[{
2552 \gdef\@charrb}]
2553 ]% }brace matching
2554
2555
2556 \message{env. counters,}
2557 %
2558 % ***** ENVIRONMENT COUNTER MACROS *****
2559 %
2560 %
2561 % An environment foo has an associated counter defined by the
2562 % following control sequences:
2563 % \c@foo : Contains the counter's numerical value. It is defined by
2564 % \newcount\foocounter.
2565 % \thefoo : Macro that expands to the printed value of \foocounter.
2566 % For example, if sections are numbered within chapters,
2567 % and section headings look like
2568 % Section II-3. The Nature of Counters
2569 % then \thesection might be defined by:
2570 % \def\thesection{\@Roman{\c@chapter}-\@arabic{\c@section}}
2571 %
2572 % \p@foo : Macro that expands to a printed 'reference prefix' of
2573 % counter foo. Any \ref to a value created by counter
2574 % foo will produce the expansion of \p@foo\thefoo when the
2575 % the \label command is executed.
2576 %
2577 % NOTE: \thefoo and \p@foo MUST BE DEFINED IN SUCH A WAY THAT
2578 % \edef\bar{\thefoo} OR \edef\bar{\p@foo}
2579 % DEFINES \bar SO THAT IT WILL EVALUATE TO THE COUNTER VALUE AT THE TIME
2580 % OF THE \edef, EVEN AFTER \foocounter AND ANY OTHER COUNTERS HAVE BEEN
2581 % CHANGED. THIS WILL HAPPEN IF YOU USE THE STANDARD COMMANDS \@arabic,
2582 % \@Roman, ETC.
2583 %
2584 % \cl@foo : List of counters to be reset when foo stepped. Has format
2585 % \@elt{countera}\@elt{counterb}\@elt{counterc}.
2586 %
2587 % The following commands are used to define and modify counters.
2588 % \setcounter{FOO}{VAL} : Globally sets \foocounter equal to VAL.

```

```

2589 % \addtocounter{FOO}{VAL}: Globally increments \foocounter by VAL.
2590 % \newcounter{NEWCTR}[OLDCTR] : Defines NEWCTR to be a counter, which is
2591 % reset when counter OLDCTR is stepped. If
2592 % NEWCTR already defined produces 'c@NEWCTR
2593 % already defined' error.
2594 % \value{CTR} : produces the value of counter CTR, for use with
2595 % a \setcounter or \addtocounter command.
2596 % \stepcounter{FOO} : Globally increments counter \c@FOO
2597 % and resets all subsidiary counters.
2598 % \refstepcounter{FOO} : Same a \stepcounter, but it also defines
2599 % \@currentreference so that a subsequent
2600 % \label{bar} command causes \ref{bar} to
2601 % generate the current value of counter foo.
2602 % \@definecounter{FOO} : Initializes counter FOO (with empty reset list),
2603 % defines \p@FOO and \theFOO to be null.
2604 % Also adds FOO to \cl@ckpt -- the reset
2605 % list of a dummy counter @ckpt used for
2606 % taking checkpoints.
2607 % \@addtoreset{FOO}{BAR} : Adds counter FOO to the list of counters
2608 % \cl@BAR to be reset when counter bar is stepped.
2609 %
2610 % NUMBERING MACROS:
2611 % \arabic{COUNTER} : Representation of COUNTER as arabic numerals.
2612 % Changed 29 Apr 86 to make it print the obvious thing
2613 % it COUNTER not positive.
2614 %
2615 % \roman{COUNTER} : Representation of COUNTER as lower-case
2616 % Roman numerals.
2617 % \Roman{COUNTER} : Representation of COUNTER as upper-case
2618 % Roman numerals.
2619 % \alph{COUNTER} : Representation of COUNTER as a lower-case
2620 % letter: 1 = a, 2 = b, etc.
2621 % \Alph{COUNTER} : Representation of COUNTER as an upper-case
2622 % letter: 1 = A, 2 = B, etc.
2623 % \fnsymbol{COUNTER} : Representation of COUNTER as a footnote
2624 % symbol: 1 = *, 2 = \dagger, etc. Can be
2625 % used only in math mode.
2626 %
2627 % THE ABOVE ARE IMPLEMENTED IN TERMS OF THE FOLLOWING:
2628 % \@arabic\FOOcounter : Representation of \FOOcounter as arabic numerals.
2629 % \@roman\FOOcounter : Representation of \FOOcounter as lower-case
2630 % Roman numerals.
2631 % \@Roman\FOOcounter : Representation of \FOOcounter as upper-case
2632 % Roman numerals.
2633 % \@alph\FOOcounter : Representation of \FOOcounter as a lower-case
2634 % letter: 1 = a, 2 = b, etc.
2635 % \@Alph\FOOcounter : Representation of \FOOcounter as an upper-case
2636 % letter: 1 = A, 2 = B, etc.
2637 % \@fnsymbol\FOOcounter : Representation of \FOOcounter as a footnote
2638 % symbol. Can be used only in math mode.
2639 %
2640 \def\setcounter#1#2{\@ifundefined{c@#1}{\@nocnterr}%
2641 {\global\csname c@#1\endcsname#2\relax}}
2642 %
2643 \def\addtocounter#1#2{\@ifundefined{c@#1}{\@nocnterr}%
2644 {\global\advance\csname c@#1\endcsname #2\relax}}
2645 %
2646 \def\newcounter#1{\expandafter\@ifdefinable \csname c@#1\endcsname
2647 {\@definecounter{#1}}\@ifnextchar[{\@newctr{#1}}{}}
2648 %
2649 \def\value#1{\csname c@#1\endcsname}
2650 %
2651 \def\@newctr#1[#2]{\@ifundefined{c@#2}{\@nocnterr}{\@addtoreset{#1}{#2}}}
2652 %
2653 \def\stepcounter#1{\global\advance\csname c@#1\endcsname \@ne
2654 {\let\@elt\@stpelt \csname cl@#1\endcsname}}
2655 %
2656 \def\@stpelt#1{\global\csname c@#1\endcsname \z@}
2657 %
2658 \def\cl@ckpt{\@elt{page}}
2659 %
2660 \def\@definecounter#1{\expandafter\newcount\csname c@#1\endcsname
2661 \setcounter{#1}0 \expandafter\gdef\csname cl@#1\endcsname{}\@addtoreset
2662 {#1}{@ckpt}\expandafter\gdef\csname p@#1\endcsname{}\expandafter

```



```

2737 %
2738 % Note: \label does the right thing in terms of spacing -- i.e.,
2739 % leaving a space on both sides of it is equivalent to leaving
2740 % a space on either side.
2741 %
2742 % This is implemented as follows. A referencable counter CNT is
2743 % incremented by the command \refstepcounter{CNT}, which sets
2744 % \@currentlabel == {CNT}{eval(\p@cnt\theCNT)}. The command
2745 % \label{FOO} then writes the following on file \@auxout :
2746 % \newlabel{FOO}{{eval(\@currentlabel)}{eval(\thepage)}}
2747 %
2748 % \ref{FOO} ==
2749 % BEGIN
2750 % if \r@foo undefined
2751 % then ??
2752 % Warning: 'reference foo on page ... undefined'
2753 % else \@car \eval(\r@FOO)\@nil
2754 % fi
2755 % END
2756 %
2757 % \pageref{foo} =
2758 % BEGIN
2759 % if \r@foo undefined
2760 % then ??
2761 % Warning: 'reference foo on page ... undefined'
2762 % else \@cdr \eval(\r@FOO)\@nil
2763 % fi
2764 % END
2765 %
2766 %
2767 % RmS 91/10/25: added a few extra \reset@font,
2768 % as suggested by Bernd Raichle
2769 \def\ref#1{\@ifundefined{r@#1}{\reset@font\bf ??}\@warning
2770 {Reference '#1' on page \thepage \space
2771 undefined}}{\edef\@tempa{\@nameuse{r@#1}}\expandafter
2772 \@car\@tempa \@nil\null}}
2773
2774 \def\pageref#1{\@ifundefined{r@#1}{\reset@font\bf ??}\@warning
2775 {Reference '#1' on page \thepage \space
2776 undefined}}{\edef\@tempa{\@nameuse{r@#1}}\expandafter
2777 \@cdr\@tempa \@nil\null}}
2778
2779 \def\newlabel#1#2{\@ifundefined{r@#1}{\@warning{Label '#1' multiply
2780 defined}}\global\@namedef{r@#1}{#2}}
2781
2782 % \label and \refstepcounter changed to allow \protect'ed commands to
2783 % work properly. For example,
2784 % \def\thechapter{\protect\foo{arabic{chapter}}.\roman{section}}
2785 % will cause a \label{bar} command to define \ref{bar} to expand to
2786 % something like \foo{4.d}. Change made 20 Jul 88.
2787
2788 \def\label#1{\@bsphack\if@filesw {\let\thepage\relax
2789 \def\protect{\noexpand\noexpand\noexpand}%
2790 \edef\@tempa{\write\@auxout{\string
2791 \newlabel{#1}{{\@currentlabel}{\thepage}}}}%
2792 \expandafter\@tempa
2793 \if@nobreak \ifvmode\nobreak\fi\fi\fi\@esphack}
2794
2795 \def\refstepcounter#1{\stepcounter{#1}\let\@tempa\protect
2796 \def\protect{\noexpand\protect\noexpand}%
2797 \edef\@currentlabel{\csname p@#1\endcsname\csname the#1\endcsname}%
2798 \let\protect\@tempa}
2799
2800 \def\@currentlabel{} % For \label commands that come before any environment
2801
2802 \message{environments,}
2803 % *****
2804 % * ENVIRONMENTS *
2805 % *****
2806 %
2807 % \begin{foo} and \end{foo} are used to delimit environment foo.
2808 % \begin{foo} starts a group and calls \foo if it is defined, otherwise
2809 % it does nothing. \end{foo} checks to see that it matches the
2810 % corresponding \begin and if so, it calls \endfoo and does an

```

```

2811 % \endgroup. Otherwise, \end{foo} does nothing.
2812 %
2813 % If \end{foo} needs to ignore blanks after it, then \endfoo should
2814 % globally set the @ignore switch true with \global\@ignoretrue.
2815 %
2816 % \@currentvir : the name of the current environment. Initialized to
2817 % 'document' to make \end{document} work right.
2818 %
2819 % \@preamblecmds : a list of commands that can be used only in the
2820 % preamble (before the \begin{document}), in the
2821 % form \do \CMDA \do \CMDB ... . These commands
2822 % are redefined to \@notprerr by \begin{document}
2823 % to save space. They include the following:
2824 % \document \documentstyle \@documentstyle
2825 % \@options \@preamblecmds \@optionlist
2826 % \@optionfiles \nofiles \includeonly \makeindex
2827 % \makeglossary
2828 % The document style can add any other commands to
2829 % this list by
2830 % \def\do{\noexpand\do\noexpand}
2831 % \edef\@preamblecmds{\@preamblecmds \do ...}
2832 %
2833 % NOTE: \@@end is defined to be the \end command of TeX82.
2834 %
2835 % \enddocument is the user's command for ending the manuscript file.
2836 %
2837 % \stop is a panic button -- to end TeX in the middle.
2838 %
2839 % \enddocument ==
2840 % BEGIN
2841 % \@checkend{document} %% checks for unmatched \begin
2842 % \clearpage
2843 % \begingroup
2844 %   if @filesw = true
2845 %     then close file @mainaux
2846 %         \global \@namedef {ARG1}{ARG2} == null
2847 %         \newlabel{LABEL}{VAL} ==
2848 %             BEGIN
2849 %                 \@tempa == VAL
2850 %                 if def(\@tempa) = def(\r@LABEL)
2851 %                     else @tempswa := true         fi
2852 %             END
2853 %         \bibcite{LABEL}{VAL} == null
2854 %         BEGIN
2855 %             \@tempa == VAL
2856 %             if def(\@tempa) = def(\g@LABEL)
2857 %                 else @tempswa := true         fi
2858 %         END
2859 %         @tempswa := false
2860 %         make @ a letter
2861 %         \input \jobname.AUX
2862 %         if @tempswa = true
2863 %             then LaTeX Warning: 'Label may have changed.
2864 %                                     Rerun to get cross-references right.'
2865 %         fi         fi
2866 % \endgroup
2867 % finish up
2868 % END
2869 %
2870 % \@writefile{EXT}{ENTRY} ==
2871 %   if tf@EXT undefined
2872 %     else \write\tf@EXT{ENTRY}
2873 %   fi
2874 %
2875 % \def\@currentvir{document}
2876 %
2877 % \def\@preamblecmds{\do\document \do\documentstyle \do\@documentstyle
2878 % \do\@options \do\@preamblecmds \do\@optionlist \do\@optionfiles
2879 % \do\@nofiles \do\includeonly \do\makeindex \do\makeglossary}
2880 %
2881 % \newif\if@ignore
2882 %
2883 % \def\enddocument{\@checkend{document}\clearpage\begingroup
2884 % \if@filesw \immediate\closeout\@mainaux

```

```

2885 \def\global\@namedef##1##2{}\def\newlabel{\@testdef r}%
2886 \def\bibcite{\@testdef b}\@tempswafalse \makeatletter\input \jobname.aux
2887 \if@tempswa \@warning{Label(s) may have changed. Rerun to get
2888 cross-references right}\fi\fi\endgroup\deadcycles\z@\@end}
2889
2890 \def\@testdef #1#2#3{\def\@tempa{#3}\expandafter \ifx \csname #1@#2\endcsname
2891 \@tempa \else \@tempwatrue \fi}
2892
2893 \long\def\@writefile#1#2{\@ifundefined{tf@#1}{}{%
2894 \immediate\write\csname tf@#1\endcsname{#2}}%
2895 % \long added 8 Feb 90, as suggested by Chris Rowley
2896
2897 \def\stop{\clearpage\deadcycles\z@\let\par\@par\@end}
2898
2899 \everypar{\@nodocument} %% To get an error if text appears before the
2900 \nullfont %% \begin{document}
2901
2902 % \begin, \end, and \@checkend changed so \end{document} will catch
2903 % an unmatched \begin. Changed 24 May 89 as suggested by
2904 % Frank Mittelbach and Rainer Sch\"opf.
2905
2906
2907 % \begin{NAME} ==
2908 % BEGIN
2909 % IF \NAME undefined THEN \@tempa == BEGIN report error END
2910 % ELSE \@tempa == (\@currenvir :=L NAME) \NAME
2911 % FI
2912 % @ignore :=G F %% Added 30 Nov 88
2913 % \begingroup
2914 % \@currenvir :=L NAME
2915 % \NAME
2916 % END
2917
2918 % \end{NAME} ==
2919 % BEGIN
2920 % \endNAME
2921 % \@checkend{NAME}
2922 % IF @endpe = T %% @endpe set True by \endparenu
2923 % THEN \@gtempa :=G \doendpe %% \doendpe redefines \par and \everypar
2924 % ELSE \@gtempa :=G \relax %% to suppress paragraph indentation in
2925 % FI %% immediately following text
2926 % \endgroup
2927 % \@gtempa
2928 % IF @ignore = T
2929 % THEN @ignore :=G F
2930 % \ignorespaces
2931 % FI
2932 % END
2933
2934 % \@checkend{NAME} ==
2935 % BEGIN
2936 % IF \@currenvir = NAME
2937 % ELSE \@badend{NAME}
2938 % FI
2939 % END
2940
2941 %% RmS 92/03/18: changed \@ignoretrue to \@ignorefalse (as documented)
2942 \def\begin#1{\@ifundefined{#1}{\def\@tempa{\@latexerr{Environment #1
2943 undefined}\@eha}}{\def\@tempa{\def\@currenvir{#1}%
2944 \csname #1\endcsname}}\global\@ignorefalse %% \global... added 2 May 90
2945 \begingroup\@endpefalse\@tempa}
2946
2947 \def\end#1{\csname end#1\endcsname\@checkend{#1}%
2948 \expandafter\endgroup \if@endpe \doendpe \fi
2949 \if@ignore \global\@ignorefalse \ignorespaces\fi}
2950
2951 \def\@checkend#1{\def\@tempa{#1}\ifx
2952 \@tempa@currenvir \else\@badend{#1}\fi}
2953
2954
2955 \message{math,}
2956 % *****
2957 % * MATH ENVIRONMENTS *
2958 % *****

```

```

2959 %
2960 % \ ( == BEGIN if math mode
2961 %         then error: '\( in math mode'
2962 %         else $
2963 %         fi
2964 %     END
2965 %
2966 % \) == BEGIN if math mode
2967 %         then if inner mode
2968 %             then $
2969 %             else error '\[ closed with \)'
2970 %         else error 'unmatched \)'
2971 %         fi
2972 %     END
2973 %
2974 % \[ == BEGIN if math mode
2975 %         then error: '\[ in math mode'
2976 %         else $$
2977 %         fi
2978 %     END
2979 %
2980 % \] == BEGIN if math mode
2981 %         then if inner mode
2982 %             then error '\( closed with \)'
2983 %             else $$
2984 %         else error 'unmatched \]'
2985 %         fi
2986 %     END
2987 %
2988 % \equation == BEGIN \refstepcounter{equation} $$ END
2989 %
2990 % \endequation == BEGIN \eqno (\theequation) $$\ignorespaces END
2991 %
2992 % NOTE: The document style must define \theequation etc., and do
2993 % the appropriate \@addtoreset. It should also redefine \@eqnnum
2994 % if another format for the equation number is desired other than the
2995 % standard (...), or to move the equation numbers to the flushleft.
2996 % (See comment on the \def of \@eqnnum.)
2997 %
2998 % \stackrel{TOP}{BOT} == PLAIN TeX's \buildrel {TOP} \over {BOT}
2999 %
3000 % \frac{TOP}{BOT} == {TOP \over BOT}
3001 %
3002 % \sqrt[N]{EXP} produces an Nth root of EXP formula.
3003 %
3004 % \: == \> (medium space)
3005 %
3006 \def\({\relax\ifmmode\@badmath\else%%$BRACE MATCH HACK
3007 \fi}
3008 %
3009 \def\){\relax\ifmmode\ifinner$\else\@badmath%%$ BRACE MATCH HACK
3010 \fi\else \@badmath\fi}
3011 %
3012 \def\[[{\relax\ifmmode\@badmath\else
3013 \ifvmode \nointerlineskip \makebox[.6\linewidth]\fi$$$$ BRACE MATCH HACK
3014 \fi}
3015 %
3016 \def\][{\relax\ifmmode\ifinner\@badmath\else$$\fi%%$ BRACE MATCH HACK
3017 \else \@badmath \fi\ignorespaces}
3018 %
3019 \let\math=\(
3020 \let\endmath=\)
3021 \def\displaymath{\[ ]
3022 \def\enddisplaymath{\]\global\@ignoretrue}
3023 %
3024 \@definecounter{equation}
3025 \def\equation{$$ % $$ BRACE MATCHING HACK
3026 \refstepcounter{equation}}
3027 %
3028 %% RmS 92/01/10: put \hbox around \@eqnnum to typeset the equation
3029 %% number in text mode (as in the eqnarray env.).
3030 \def\endequation{\eqno \hbox{\@eqnnum}% $$ BRACE MATCHING HACK
3031 $$\global\@ignoretrue}
3032

```

```

3033 % \@eqnnum: Produces the equation number for equation and
3034 %   eqnarray environments. The following definition is for
3035 %   flushright numbers; for flushleft numbers, see legno.doc.
3036 %   The {\rm ... } puts the equation number in roman type even if
3037 %   an eqnarray environment appears in an italic environment.
3038 %
3039 %% RmS 91/09/29: \reset@font added.
3040 \def\@eqnnum{\reset@font\rm (\theequation)}
3041
3042
3043 \def\stackrel#1#2{\mathrel{\mathop{#2}\limits^{#1}}}
3044 \def\frac#1#2{{#1\over #2}}
3045
3046 \let\@sqrt=\sqrt
3047 \def\sqrt{\@ifnextchar[{\@sqrt}{\@sqrt}}
3048 \def\sqrt[#1]{\root #1\of}
3049
3050 \let\:=\>
3051
3052 % Here's the eqnarray environment:
3053 % Default is for left-hand side of equations to be flushleft.
3054 % To make them flushright, \let\@eqnrel = \hfil
3055
3056 \newcount\@eqcnt
3057 \newcount\@eqpen
3058 \newif\if@eqnsw\@eqnswtrue
3059
3060 \@centering = 0pt plus 1000pt % Changed 11/4/85 to produce warning message
3061 % if line extends into margin. Doesn't warn
3062 % about formula overprinting equation number.
3063
3064 \def\eqnarray{\stepcounter{equation}\let\@currentlabel\theequation
3065 \global\@eqnswtrue\m@th
3066 \global\@eqcnt\z@\tabskip\@centering\let\\\@eqnrcr
3067 $$$\halign to\displaywidth\bgroup\@eqnrel\hskip\@centering
3068   $\displaystyle\tabskip\z@{##}$$\global\@eqcnt\@ne
3069   \hskip 2\arraycolsep \hfil${##}$\hfil
3070   &\global\@eqcnt\tw@ \hskip 2\arraycolsep $\displaystyle\tabskip\z@{##}$\hfil
3071   \tabskip\@centering&\llap{##}\tabskip\z@\cr}
3072
3073 \def\endeqnarray{\@@eqnrcr\egroup
3074   \global\advance\c@equation\m@ne$$\global\@ignoretrue}
3075
3076 \let\@eqnrel=\relax
3077
3078 \def\nonumber{\global\@eqnswfalse}
3079
3080 \def\@eqnrcr{\@ifnum0='}\fi\@ifstar{\global\@eqpen\@M
3081   \@yeqnrcr}{\global\@eqpen\interdisplaylinepenalty \@yeqnrcr}}
3082
3083 \def\@yeqnrcr{\@ifnextchar [{\@xeqnrcr}{\@xeqnrcr[\z@]}}
3084
3085 \def\@xeqnrcr[#1]{\ifnum0='{ \fi}\@@eqnrcr
3086   \noalign{\penalty\@eqpen\vskip\jot\vskip #1\relax}}
3087
3088 \def\@@eqnrcr{\let\@tempa\relax
3089   \ifcase\@eqcnt \def\@tempa{& & }\or \def\@tempa{& & }%
3090   \else \def\@tempa{& }\fi
3091   \@tempa \if@eqnsw\@eqnnum\stepcounter{equation}\fi
3092   \global\@eqnswtrue\global\@eqcnt\z@\cr}
3093
3094 % Here's the eqnarray* environment:
3095
3096 \let\@seqnrcr=\@eqnrcr
3097 \@namedef{eqnarray*}{\def\@eqnrcr{\nonumber\@seqnrcr}\eqnarray}
3098 \@namedef{endeqnarray*}{\nonumber\endeqnarray}
3099
3100 % \lefteqn{FORMULA} typesets FORMULA in display math style
3101 % flushleft in a box of width zero.
3102 %
3103
3104 \def\lefteqn#1{\hbox to\z@{\displaystyle #1$\hss}}
3105
3106

```

```

3107 \message{center,}
3108 % *****
3109 % *      CENTER, FLUSHRIGHT, FLUSHLEFT, ETC. *
3110 % *****
3111 %
3112 %
3113 % \center, \flushright and \flushleft set
3114 % \rightskip = Opt or \@flushglue (as appropriate)
3115 % \leftskip = Opt or \@flushglue (as appropriate)
3116 % \parindent = Opt
3117 % \parfillskip = Opt. (except \flushleft)
3118 % ||      == \par \vskip -\parskip
3119 % |[LENGTH] == || \vskip LENGTH
3120 % ||*     == \par \penalty 10000 \vskip -\parskip
3121 % |[LEN]  == |[* \vskip LENGTH
3122 %
3123 % They invoke the trivlist environment to handle vertical spacing before
3124 % and after them.
3125 %
3126 % \centering, \raggedright and \raggedleft are the declaration analogs
3127 % of the above.
3128 %
3129 % \raggedright has a more universal effect, however. It sets
3130 % \@rightskip := flushglue. Every environment, like the list environments,
3131 % that set \rightskip to its 'normal' value set it to \@rightskip
3132 %
3133 \def\@centercr{\ifhmode \unskip\else \@badcrerr\fi
3134 \par\@ifstar{\penalty \@M\@xcentercr}{\@xcentercr}}
3135 %
3136 \def\@xcentercr{\addvspace{-\parskip}\@ifnextchar
3137 [ {\@icentercr}{\ignorespaces}}
3138 %
3139 \def\@icentercr[#1]{\vskip #1\ignorespaces}
3140 %
3141 \def\center{\trivlist \centering\item[]}
3142 \def\centering{\let\@=\@centercr\rightskip\@flushglue\leftskip\@flushglue
3143 \parindent\z@\parfillskip\z@}
3144 \let\endcenter=\endtrivlist
3145 %
3146 \newskip\@rightskip \@rightskip \z@
3147 %
3148 \def\flushleft{\trivlist \raggedright\item[]}
3149 \def\raggedright{\let\@=\@centercr\rightskip\@flushglue \rightskip\@rightskip
3150 \leftskip\z@
3151 \parindent\z@}
3152 \let\endflushleft=\endtrivlist
3153 %
3154 \def\flushright{\trivlist \raggedleft\item[]}
3155 \def\raggedleft{\let\@=\@centercr\rightskip\z@\leftskip\@flushglue
3156 \parindent\z@\parfillskip\z@}
3157 \let\endflushright=\endtrivlist
3158 %
3159 \message{verbatim,}
3160 % *****
3161 % *      VERBATIM *
3162 % *****
3163 %
3164 % The verbatim environment uses the fixed-width \tt font, turns blanks into
3165 % spaces, starts a new line for each carriage return (or sequence of
3166 % consecutive carriage returns), and interprets EVERY character literally.
3167 % I.e., all special characters \, {, $, etc. are \catcode'd to 'other'.
3168 %
3169 % The command \verb produces in-line verbatim text, where the argument
3170 % is delimited by any pair of characters. E.g., \verb #...# takes
3171 % '...' as its argument, and sets it verbatim in \tt font.
3172 %
3173 % The *-variants of these commands is the same, except that spaces
3174 % print as the TeXbook's space character instead of as blank spaces.
3175 %
3176 {\catcode'\^M=13 \gdef\@gobblecr{\@ifnextchar
3177 {\@gobble}{\ignorespaces}}}
3178 %
3179 {\catcode'\ =\active\gdef\@vobeyspaces{\catcode'\ \active\let \@xobeysp}}
3180 %

```

```

3181 % Definition of \@xobeysp changed on 19 Nov 86 from
3182 % \def\@xobeysp{\leavevmode{ } }
3183 % to prevent line breaks at spaces. Change suggested by
3184 % Nelson Beebe
3185 %
3186 \def\@xobeysp{\leavevmode\penalty10000\ }
3187
3188
3189
3190 \begingroup \catcode '|=0 \catcode '= 1
3191 \catcode'=2 \catcode '{=12 \catcode '\}=12
3192 \catcode'\}=12 |gdef|\@xverbatim#1\end{verbatim}[#1\end[verbatim]]
3193 |gdef|\@sxverbatim#1\end{verbatim*}[#1\end[verbatim*]]
3194 |endgroup
3195
3196 % \@sverbatim obsolete -- removed 24 May 89, as suggested by
3197 % Rainer Sch\"opf and Frank Mittelbach
3198 % \def\@sverbatim{\obeyspaces\@verbatim}
3199
3200 \def\@gobble#1{}
3201
3202 % 91/07/24 RmS: added \penalty\interlinepenalty to definition
3203 % of \par so that \samepage works.
3204
3205 \def\@verbatim{\trivlist \item[]\if@minipage\else\vskip\parskip\fi
3206 \leftskip\@totalleftmargin\rightskip\z@
3207 \parindent\z@\parfillskip\@flushglue\parskip\z@
3208 %%RmS 91/08/26 Added \@par to clear possible \parshape definition
3209 %%from a surrounding list (the verbatim guru says)
3210 \@par
3211 \@tempswafalse \def\par{\if@tempswa\hbox{} \fi \@tempswattrue \@par
3212 \penalty\interlinepenalty}%
3213 \obeyspaces \tt \catcode'=13 \@noligs \let\do\@makeother \dospecials}
3214
3215 \def\verbatim{\@verbatim \frenchspacing\@vobeyspaces \@xverbatim}
3216 \let\endverbatim=\endtrivlist
3217
3218 \@namedef{verbatim*}{\@verbatim\@sxverbatim}
3219 \expandafter\let\csname endverbatim*\endcsname =\endtrivlist
3220
3221
3222 \def\@makeother#1{\catcode'#1=12\relax}
3223
3224 \def\verb{\begingroup \catcode'=13 \@noligs
3225 \tt \let\do\@makeother \dospecials
3226 \@ifstar{\@sverb}{\@verb}}
3227
3228 % Definitions of \@sverb and \@verb changed so \verb+ foo+ does not lose
3229 % leading blanks when it comes at the beginning of a line.
3230 % Change made 24 May 89. Suggested by Frank Mittelbach and Rainer Sch\"opf.
3231 %
3232 \def\@sverb#1{\def\@tempa ##1#1{\leavevmode\null##1\endgroup}\@tempa}
3233
3234 \def\@verb{\@vobeyspaces \frenchspacing \@sverb}
3235
3236
3237 %% \@noligs prevents ?' and !' from being treated as ligatures
3238 %% added 19 April 86
3239
3240 \begingroup
3241 \catcode'=13
3242 \gdef\@noligs{\let'\@lquote}
3243 \endgroup
3244
3245 %% RmS 91/06/21: added \leavevmode to definition of \@lquote
3246 %% to avoid the \kern being processed in vertical mode
3247
3248 \def\@lquote{\leavevmode{\kern\z@}' }
3249 \message{list,}
3250 % *****
3251 % * THE LIST ENVIRONMENT *
3252 % *****
3253 %
3254 % The generic commands for creating an indented environment -- enumerate,

```

```

3255 % itemize, quote, etc -- are
3256 %   \list{LABEL}{COMMANDS} ... \endlist
3257 % which can be invoked by the user as the list environment. The LABEL
3258 % argument specifies item labeling. COMMANDS contains commands for
3259 % changing the horizontal and vertical spacing parameters.
3260 %
3261 % Each item of the environment is begun by the command \item[ITEMLABEL]
3262 % which produces an item labeled by ITEMLABEL. If the argument is
3263 % missing, then the LABEL argument of the \list command is used as the
3264 % item label.
3265 %
3266 % The label is formed by putting \makelabel{ITEMLABEL} in an hbox whose
3267 % width is either its natural width or else \labelwidth, whichever is
3268 % larger. The \list command defines \makelabel to have the default
3269 % definition
3270 %   \makelabel{ARG} == BEGIN \hfil ARG END
3271 % which, for a label of width less than \labelwidth, puts the label
3272 % flushright, \labelsep to the left of the item's text. However,
3273 % \makelabel can be \let to another command by the \list's COMMANDS
3274 % argument.
3275 %
3276 % A \usecounter{foo} command in the second argument causes the counter
3277 % foo to be initialized to zero, and stepped by every \item command
3278 % without an argument. (\label commands within the list refer to this
3279 % counter.)
3280 %
3281 % When you leave a list environment, returning either to an enclosing
3282 % list or normal text mode, LaTeX begins a new paragraph if and only if
3283 % you leave a blank line after the \end command. This is accomplished
3284 % by the \@endparenu command.
3285 %
3286 % Blank lines are ignored every other reasonable place--i.e.:
3287 %   - Between the \begin{list} and the first \item,
3288 %   - Between the \item and the text of that item.
3289 %   - Between the end of the last item and the \end{list}.
3290 %
3291 % For an environment like quotation, in which items are not labeled,
3292 % the entire environment is a single item. It is defined by
3293 % letting \quotation == \list{}{...}\item[]. (Note the [], there in
3294 % case the first character in the environment is a '['.) The spacing
3295 % parameters provide a great deal of flexibility in designing the
3296 % format, including the ability to let the indentation of the first
3297 % paragraph be different from that of the subsequent ones.
3298 %
3299 % The trivlist environment is equivalent to a list environment
3300 % whose second argument sets the following parameter values:
3301 %   \leftmargin = 0 : causes no indentation of left margin
3302 %   \labelwidth = 0 : see below for precise effect this has.
3303 %   \itemindent = 0 : with a null label, makes first paragraph
3304 %   have no indentation. Succeeding paragraphs have \parindent
3305 %   indentation. To give first paragraph same indentation, set
3306 %   \itemindent = \parindent before the \item[].
3307 % Every \item in a trivlist environment must have an argument---in many
3308 % cases, this will be the null argument (\item[]). The trivlist
3309 % environment is mainly used for paragraphing environments, like
3310 % verbatim, in which there is no margin change. It provides the same
3311 % vertical spacing as the list environment, and works reasonably well
3312 % when it occurs immediately after an \item command in an enclosing list.
3313 %
3314 % The following variables are used inside a list environment:
3315 %   \@totalleftmargin : The distance that the prevailing left margin is
3316 %   indented from the outermost left margin,
3317 %   \linewidth       : The width of the current line. Must be
3318 %   initialized to \hsize.
3319 %   \@listdepth      : A count for holding current list nesting depth.
3320 %   \makelabel       : A macro with a single argument, used to generate
3321 %   the label from the argument (given or implied) of the
3322 %   \item command. Initialized to \@mklab by the \list
3323 %   command. This command must produce some stretch--i.e.,
3324 %   an \hfil.
3325 %   @inlabel         : A switch that is false except between the time an
3326 %   \item is encountered and the time that TeX actually
3327 %   enters horizontal mode. Should be tested by
3328 %   commands that can be messed up by the list

```



```

3329 % environment's use of \everypar.
3330 % \box\@labels : When @inlabel = true, it holds the labels
3331 % to be put out by \everypar.
3332 % @noparitem : A switch set by \list when @inlabel = true.
3333 % Handles the case of a \list being the first thing
3334 % in an item.
3335 % @noparlist : A switch set true for a list that begins an
3336 % item. No \topsep space is added before or after
3337 % such a list.
3338 % @newlist : Set true by \list, set false by the first \item's
3339 % text (by \everypar).
3340 % @noitemarg : Set true when executing an \item with no explicit
3341 % argument. Used to save space. To save time,
3342 % make two separate \@item commands.
3343 % @nmbriest : Set true by \usecounter command, causes list to
3344 % be numbered.
3345 % \@listctr : \def'ed by \usecounter to name of counter.
3346 % @noskipsec : A switch set true by a sectioning command when it is
3347 % creating an in-text heading with \everypar.
3348 %
3349 % Throughout a list environment, \hspace is the width of the current
3350 % line, measured from the outermost left margin to the outermost right
3351 % margin. Environments like tabbing should use \linewidth instead of
3352 % \hspace.
3353 %
3354 % Here are the parameters of a list that can be set by commands in
3355 % the \list's COMMANDS argument. These parameters are all TeX
3356 % skips or dimensions (defined by \newskip or \newdimen), so the usual
3357 % TeX or LaTeX commands can be used to set them. The commands will
3358 % be executed in vmode if and only if the \list was preceded by a
3359 % \par (or something like an \end{list}), so the spacing parameters
3360 % can be set according to whether the list is inside a paragraph
3361 % or is its own paragraph.
3362 %
3363 % VERTICAL SPACING (skips):
3364 %
3365 % \topsep : Space between first item and preceding paragraph.
3366 % \partopsep : Extra space added to \topsep when environment starts
3367 % a new paragraph (is called in vmode).
3368 % \itemsep : Space between successive items.
3369 % \parsep : Space between paragraphs within an item -- the \parskip
3370 % for this environment.
3371 %
3372 % PENALTIES
3373 % \@beginparpenalty : put at the beginning of a list
3374 % \@endparpenalty : put at end of list
3375 % \@itempenalty : put between items.
3376 %
3377 % HORIZONTAL SPACING (dimens)
3378 % \leftmargin : space between left margin of enclosing environment
3379 % (or of page if top level list) and left margin of
3380 % this list. Must be nonnegative.
3381 % \rightmargin : analogous.
3382 % \listparindent : extra indentation at beginning of every paragraph
3383 % of a list except the one started by the \item
3384 % command. May be negative! Usually, labeled lists
3385 % have \listparindent equal to zero.
3386 % \itemindent : extra indentation added right BEFORE an item label.
3387 % \labelwidth : nominal width of box that contains the label.
3388 % If the natural width of the label <= \labelwidth,
3389 % then the label is flushed right inside a box
3390 % of width \labelwidth (with an \hfil). Otherwise,
3391 % a box of the natural width is employed, which causes
3392 % an indentation of the text on that line.
3393 % \labelsep : space between end of label box and text of
3394 % first item.
3395 %
3396 % DEFAULT VALUES:
3397 % Defaults for the list environment are set as follows.
3398 % First, \rightmargin, \listparindent and \itemindent are set
3399 % to Opt. Then, one of the commands \@listi, \@listii, ... , \@listvi
3400 % is called, depending upon the current level of the list.
3401 % The \@list... commands should be defined by the document
3402 % style. A convention that the document style should follow is

```

```

3403 %   to set \leftmargin to \leftmargini, ... , \leftmarginvi for
3404 %   the appropriate level. Items that aren't changed may be left
3405 %   alone, but everything that could possibly be changed must be
3406 %   reset.
3407 %
3408 % \list[LABEL]{COMMANDS} ==
3409 % BEGIN
3410 %   if \@listdepth > 5
3411 %     then LaTeX error: 'Too deeply nested'
3412 %     else \@listdepth :=G \@listdepth + 1
3413 %   fi
3414 %   \rightmargin := Opt
3415 %   \listparindent := Opt
3416 %   \itemindent := Opt
3417 %   \eval{@list \romannumeral\the\@listdepth} %% Set default values:
3418 %   \@itemlabel :=L LABEL
3419 %   \makelabel == \@mklab
3420 %   @nmbriest :=L false
3421 %   COMMANDS
3422 %
3423 %   \@trivlist           % commands common to \list and \trivlist
3424 %
3425 %   \parskip :=L \parsep
3426 %   \parindent :=L \listparindent
3427 %   \linewidth :=L \linewidth - \rightmargin - \leftmargin
3428 %   \@totalleftmargin :=L \@totalleftmargin + \leftmargin
3429 %   \parshape 1 \@totalleftmargin \linewidth
3430 %   \ignorespaces           % gobble space up to \item
3431 % END
3432 %
3433 % \endlist == BEGIN \@listdepth :=G \@listdepth -1
3434 %             \endtrivlist
3435 %             END
3436 %
3437 % \@trivlist ==
3438 % BEGIN
3439 %   if @newlist = T then \@noitemerr fi %% This command removed for some
3440 %                                     %% forgotten reason.
3441 %   \@topsepadd :=L \topsep
3442 %   if @noskipsec then leave vertical mode fi %% Added 11 Jun 85
3443 %   if vertical mode
3444 %     then \@topsepadd :=L \@topsepadd + \partopsep
3445 %     else \unskip \par           % remove glue from end of last line
3446 %   fi
3447 %   if @inlabel = true
3448 %     then @nparitem :=L true
3449 %     @nparlist :=L true
3450 %     else @nparlist :=L false
3451 %     \@topsep :=L \@topsepadd
3452 %   fi
3453 %   \@topsep :=L \@topsep + \parskip %% Change 4 Sep 85
3454 %   \leftskip :=L Opt           % Restore paragraphing parameters
3455 %   \rightskip :=L \@rightskip
3456 %   \parfillskip :=L Opt + 1fil
3457 %
3458 %   NOTE: \@setpar called on every \list in case \par has been temporarily
3459 %         munged before the \list command.
3460 %   \@setpar{if @newlist = false then {\@@par} fi}
3461 %   \@newlist :=G T
3462 %   \@outerparskip :=L \parskip
3463 % END
3464 %
3465 % \trivlist ==
3466 % BEGIN
3467 %   \parsep := \parskip
3468 %   \@trivlist
3469 %   \labelwidth := 0
3470 %   \leftmargin := 0
3471 %   \itemindent := \parindent
3472 %   \makelabel[LABEL] == LABEL
3473 % END
3474 %
3475 % \endtrivlist ==
3476 % BEGIN

```

```

3477 %   if @inlabel = T then \indent fi
3478 %   if horizontal mode then \unskip \par fi
3479 %   if @noparlist = true
3480 %       else if \lastskip > 0
3481 %           then \@tempkipa := \lastskip
3482 %               \vskip -\lastskip
3483 %               \vskip \@tempkipa -\@outerparskip + \parskip
3484 %           fi
3485 %       \@endparenu
3486 %   fi
3487 % END
3488 %
3489 % \@endparenu ==
3490 % BEGIN
3491 %   \addpenalty{@endparpenalty}
3492 %   \addvspace{@topsepadd}
3493 %   \endgroup %% ends the \begin command's \begin group
3494 %   \par == BEGIN
3495 %       \@restorepar
3496 %       \everypar{}
3497 %       \par
3498 %   END
3499 %   \everypar == BEGIN remove \lastbox \everypar{} END
3500 %   \begingroup %% to match the \end commands \endgroup
3501 % END
3502 %
3503 % \item == BEGIN if next char = [
3504 %     then \@item
3505 %     else @noitemarg := true
3506 %         \@item[@itemlabel]
3507 %     END
3508 %
3509 % \@item[LAB] ==
3510 % BEGIN
3511 %   if @noparitem = true
3512 %       then @noparitem := false
3513 %           % NOTE: then clause
3514 %           % hardly every taken,
3515 %           \box\@labels :=G \hbox{\hskip -\leftmargin % so made a macro
3516 %               \box\@labels % \donoparitem
3517 %               \hskip \leftmargin }
3518 %       if @minipage = false then
3519 %           \@tempkipa := \lastskip
3520 %           \vskip -\lastskip
3521 %           \vskip \@tempkipa + \@outerparskip - \parskip
3522 %       fi
3523 %       else if @inlabel = true
3524 %           then \indent \par % previous item empty.
3525 %       fi
3526 %       if hmode then 2 \unskip's % To remove any space at end of prev.
3527 %           \par % paragraph that could cause a blank
3528 %               % line.
3529 %       fi
3530 %       if @newlist = T
3531 %           then if @nobreak = T % Kludge if list follows \section
3532 %               then \addvspace{@outerparskip - \parskip}
3533 %                   else \addpenalty{@beginparpenalty}
3534 %                       \addvspace{@topsep}
3535 %                       \addvspace{-\parskip} %% added 4 Sep 85
3536 %                   fi
3537 %           else \addpenalty{@itempenalty}
3538 %               \addvspace{@itemsep}
3539 %           fi
3540 %       @inlabel :=G true
3541 %   fi
3542 %
3543 % \everypar{ @minipage :=G F
3544 %     @newlist :=G F
3545 %     if @inlabel = true
3546 %         then @inlabel :=G false
3547 %             \hskip -\parindent
3548 %             \box\@labels
3549 %             \penalty 0 %% 3 Oct 85 -- allow line break here
3550 %             \box\@labels :=G null
3551 %         fi
3552 %     \everypar{} }

```

```

3551 % @nobreak :=G false
3552 % if @noitemarg = true
3553 % then @noitemarg := false
3554 % if @nmbrrlist
3555 % then \refstepcounter{@listctr}
3556 % fi fi
3557 % \@tempboxa :=L \hbox{\makelabel{LAB}}
3558 % \box@labels :=G \@labels \hskip \itemindent
3559 % \hskip - (\labelwidth + \labelsep)
3560 % if \wd \@tempboxa > \labelwidth
3561 % then \box@tempboxa
3562 % else \hbox to \labelwidth {\makelabel{LAB}}
3563 % fi
3564 % \hskip\labelsep
3565 % \ignorespaces %gobble space up to text
3566 % END
3567 %
3568 % \usecounter{CTR} == BEGIN @nmbrrlist :=L true
3569 % \@listctr == CTR
3570 % \setcounter{CTR}{0}
3571 % END
3572 %
3573 % DEFINE \dimen's and \count
3574 \newskip\topsep
3575 \newskip\partopsep
3576 \newskip\itemsep
3577 \newskip\parsep
3578 \newskip\@topsep
3579 \newskip\@topsepadd
3580 \newskip\@outerparskip
3581
3582 \newdimen\leftmargin
3583 \newdimen\rightmargin
3584 \newdimen\listparindent
3585 \newdimen\itemindent
3586 \newdimen\labelwidth
3587 \newdimen\labelsep
3588 \newdimen\linewidth
3589 \newdimen\@totalleftmargin \@totalleftmargin=\z@
3590 \newdimen\leftmargini
3591 \newdimen\leftmarginii
3592 \newdimen\leftmarginiii
3593 \newdimen\leftmarginiv
3594 \newdimen\leftmarginv
3595 \newdimen\leftmarginvi
3596
3597 \newcount\@listdepth \@listdepth=0
3598 \newcount\@itempenalty
3599 \newcount\@beginparpenalty
3600 \newcount\@endparpenalty
3601
3602 \newbox\@labels
3603
3604 \newif\if@inlabel \@inlabelfalse
3605 \newif\if@newlist \@newlistfalse
3606 \newif\if@nparitem \@nparitemfalse
3607 \newif\if@nparlist \@nparlistfalse
3608 \newif\if@noitemarg \@noitemargfalse
3609 \newif\if@nmbrrlist \@nmbrrlistfalse
3610
3611 \def\list#1#2{\ifnum \@listdepth >5\relax \@toodeep
3612 \else \global\advance\@listdepth\@ne \fi
3613 \rightmargin \z@ \listparindent\z@ \itemindent\z@
3614 \csname @list\romannumeral\the\@listdepth\endcsname
3615 \def\@itemlabel{#1}\let\makelabel\@mklab \@nmbrrlistfalse #2\relax
3616 \@trivlist
3617 \parskip\parsep \parindent\listparindent
3618 \advance\linewidth -\rightmargin \advance\linewidth -\leftmargin
3619 \advance\@totalleftmargin \leftmargin
3620 \parshape \@ne \@totalleftmargin \linewidth
3621 \ignorespaces}
3622
3623 \def\@trivlist{\@topsepadd\topsep
3624 \if@noskipsec \leavevmode \fi

```

```

3625 \ifvmode \advance\@topsepadd\partopsep \else \unskip\par\fi
3626 \if@inlabel \@noperitemtrue \@noperlisttrue
3627 \else \@noperlistfalse \@topsep\@topsepadd \fi
3628 \advance\@topsep \parskip
3629 \leftskip\z@\rightskip\rightskip \parfillskip\@flushglue
3630 \@setpar{\if@newlist\else{\@par}\fi}%
3631 \global\@newlisttrue \outerparskip\parskip}
3632
3633 %% RmS 92/03/18 added \@nمبرlistfalse
3634 \def\trivlist{\parsep\parskip\@nمبرlistfalse
3635 \@trivlist \labelwidth\z@ \leftmargin\z@
3636 \itemindent\z@ \def\makelabel##1{##1}}
3637
3638 \def\endlist{\global\advance\@listdepth\m@ne
3639 \endtrivlist}
3640
3641 % Definition of \endtrivlist moved earlier in file so other commands
3642 % can be \let = to it.
3643
3644 \def\@mklab#1{\hfil #1}
3645
3646 \def\item{\ifnextchar [{\@item}{\@noitemargtrue \@item[\@itemlabel]}}
3647
3648 \def\@donoparitem{\@noperitemfalse
3649 \global\setbox\@labels\hbox{\hskip -\leftmargin
3650 \unhbox\@labels
3651 \hskip \leftmargin}\if@minipage\else
3652 \@tempkipa\lastskip
3653 \vskip -\lastskip \advance\@tempkipa\@outerparskip
3654 \advance\@tempkipa -\parskip \vskip\@tempkipa\fi}
3655
3656 \def\@item[#1]{\if@noperitem \@donoparitem
3657 \else \if@inlabel \indent \par \fi
3658 \ifhmode \unskip\unskip \par \fi
3659 \if@newlist \if@nbreak \@nbitem \else
3660 \addpenalty\@beginparpenalty
3661 \addvspace\@topsep \addvspace{-\parskip}\fi
3662 \else \addpenalty\@itempenalty \addvspace\itemsep
3663 \fi
3664 \global\@inlabeltrue
3665 \fi
3666 \everypar{\global\@minipagefalse\global\@newlistfalse
3667 \if@inlabel\global\@inlabelfalse \hskip -\parindent \box\@labels
3668 \penalty\z@ \fi
3669 \everypar{}}\global\@nbreakfalse
3670 \if@noitemarg \@noitemargfalse \if@nمبرlist \refstepcounter{\@listctr}\fi \fi
3671 \setbox\@tempboxa\hbox{\makelabel{#1}}%
3672 \global\setbox\@labels
3673 \hbox{\unhbox\@labels \hskip \itemindent
3674 \hskip -\labelwidth \hskip -\labelsep
3675 \ifdim \wd\@tempboxa >\labelwidth
3676 \box\@tempboxa
3677 %% RmS 91/11/22: Changed second call to \makelabel to \unhbox\@tempboxa.
3678 %% AVOIDS PROBLEMS WITH SIDE EFFECTS IN \makelabel AND IS
3679 %% MORE EFFICIENT.
3680 % \else \hbox to\labelwidth {\makelabel{#1}}\fi
3681 \else \hbox to\labelwidth {\unhbox\@tempboxa}\fi
3682 \hskip \labelsep}\ignorespaces}
3683
3684 %% RmS 91/11/04: added default definition for \makelabel,
3685 %% to produce an error message.
3686 \def\makelabel#1{\@latexerr{Lonely \string\item--perhaps a missing
3687 list environment}\@ehc}
3688
3689 \def\@nbitem{\@tempkipa\@outerparskip \advance\@tempkipa -\parskip
3690 \addvspace{\@tempkipa}}
3691
3692 \def\usecounter#1{\@nمبرlisttrue\def\@listctr{#1}\setcounter{#1}\z@}
3693
3694 \message{itemize,}
3695 % *****
3696 % * ITEMIZE AND ENUMERATE *
3697 % *****
3698 %

```

```

3699 % Enumeration is done with four counters: enumi, enumii, enumiii
3700 % and enumiv, where enumN controls the numbering of the Nth level
3701 % enumeration. The label is generated by the commands
3702 % \labelenumi ... \labelenumiv, which should be defined by the
3703 % document style. Note that \p@enumN\theenumN defines the output
3704 % of a \ref command. A typical definition might be:
3705 %   \def\theenumii{\alph{enumii}}
3706 %   \def\p@enumii{\theenumi\theenumii}
3707 %   \def\labelenumii{\theenumii}
3708 % which will print the labels as '(a)', '(b)', ... and print a \ref as
3709 % '3a'.
3710 %
3711 % The item numbers are moved to the right of the label box, so they are
3712 % always a distance of \labelsep from the item.
3713 %
3714 % \@enumdepth holds the current enumeration nesting depth.
3715 %
3716 % Itemization is controlled by four commands: \labelitemi, \labelitemii,
3717 % \labelitemiii, and \labelitemiv. To cause the second-level list to be
3718 % bulleted, you just define \labelitemii to be $\bullet$. \@itemspacing
3719 % and \@itemdepth are the analogs of \@enumspacing and \@enumdepth.
3720 %
3721 % \enumerate ==
3722 % BEGIN
3723 %   if \@enumdepth > 3
3724 %     then errormessage: 'Too deeply nested'.
3725 %     else \@enumdepth :=L \@enumdepth + 1
3726 %         \@enumctr :=L eval(enum@romannumeral\the\@enumdepth)
3727 %         \list{\label{\@enumctr}}
3728 %             {\usecounter{\@enumctr}
3729 %              \makelabel{LABEL} == \hss \llap{LABEL}}
3730 %     fi
3731 % END
3732 %
3733 % \endenumerate == \endlist
3734 %
3735 \newcount\@enumdepth \@enumdepth = 0
3736
3737 \definecounter{enumi}
3738 \definecounter{enumii}
3739 \definecounter{enumiii}
3740 \definecounter{enumiv}
3741
3742 \def\enumerate{\ifnum \@enumdepth >3 \toodeep\else
3743   \advance\@enumdepth \@ne
3744   \edef\@enumctr{enum@romannumeral\the\@enumdepth}\list
3745   {\csname label\@enumctr\endcsname}{\usecounter
3746     {\@enumctr}\def\makelabel##1{\hss\llap{##1}}}\fi}
3747
3748 \let\endenumerate =\endlist
3749
3750
3751 % \itemize ==
3752 % BEGIN
3753 %   if \@itemdepth > 3
3754 %     then errormessage: 'Too deeply nested'.
3755 %     else \@itemdepth :=L \@itemdepth + 1
3756 %         \@itemitem == eval(labelitem\romannumeral\the\@itemdepth)
3757 %         \list{\@nameuse{\@itemitem}}
3758 %             {\makelabel{LABEL} == \hss \llap{LABEL}}
3759 %     fi
3760 % END
3761 %
3762 % \enditemize == \endlist
3763 %
3764 \newcount\@itemdepth \@itemdepth = 0
3765
3766 \def\itemize{\ifnum \@itemdepth >3 \toodeep\else \advance\@itemdepth \@ne
3767 \edef\@itemitem{labelitem\romannumeral\the\@itemdepth}%
3768 \list{\csname\@itemitem\endcsname}{\def\makelabel##1{\hss\llap{##1}}}\fi}
3769
3770 \let\enditemize =\endlist
3771
3772 \message{boxes,}

```

```

3773 % *****
3774 % * BOXES *
3775 % *****
3776 %
3777 % USER COMMANDS:
3778 %
3779 % \makebox [WID][POS]{OBJ}
3780 % : puts OBJ in an \hbox of width WID, positioned by POS.
3781 % POS = l -> flushleft, POS = r -> flushright.
3782 % Default is centered.
3783 % If WID is missing, then POS is also missing and OBJ
3784 % is put in an \hbox of its natural width.
3785 %
3786 % \mbox{OBJ} == \makebox{OBJ}, and is more efficient.
3787 %
3788 % \makebox (X,Y)[POS]{OBJ}
3789 % : puts OBJ in an \hbox of width X * \unitlength
3790 % and height Y * \unitlength. POS arguments are
3791 % l or r for flushleft, flushright and t or b
3792 % for top, bottom -- or combinations like tr or rb.
3793 % Default for horizontal and vertical are centered.
3794 %
3795 % \newsavebox{CMD} : If \CMD is undefined, then defines it
3796 % to be a TeX box register.
3797 %
3798 % \savebox {CMD} ... : \CMD is defined to be a TeX box register,
3799 % and the '...' are any \makebox arguments. It is
3800 % like \makebox, except it doesn't produce text but
3801 % saves the value in \box \CMD.
3802 % \sbox N{OBJ} is an efficient abbreviation for
3803 % \savebox N{OBJ}.
3804 %
3805 % \framebox ... : like \makebox, except it puts a 'frame' around
3806 % the box. The frame is made of lines of thickness
3807 % \fboxrule, separated by space \fboxsep from the
3808 % text -- except for \framebox(X,Y) ... , where the
3809 % thickness of the lines is as for the picture environment,
3810 % and there is no separation added.
3811 % \fbox{OBJ} is an efficient abbreviation for \framebox{OBJ}
3812 %
3813 % \parbox[POS]{WIDTH}{TEXT} : Makes a box with \hspace TEXT, positioned
3814 % by POS as follows:
3815 % c : \vcenter (placed in $...$ if not in math mode)
3816 % b : \vbox
3817 % t : \vtop
3818 % default value is c.
3819 % Sets \hspace := WIDTH and calls \@parboxrestore, which does
3820 % the following:
3821 % Restores the original definitions of:
3822 % \par
3823 % \l
3824 % \l' \l' \l' \l'
3825 % Resets the following parameters:
3826 % \parindent = Opt
3827 % \parskip = Opt %% added 20 Jan 87
3828 % \linewidth = \hspace
3829 % \@totalleftmargin = Opt
3830 % \leftskip = Opt
3831 % \rightskip = Opt
3832 % \@rightskip = Opt
3833 % \parfillskip = Opt plus 1fil
3834 % \lineskip = \normallineskip
3835 % \baselineskip = \normalbaselineskip
3836 % Calls \sloppy
3837 %
3838 % Note: \@arrayparboxrestore same as \@parboxrestore
3839 % but it doesn't restore \l.
3840 %
3841 % \minipage : Similar to parbox, except it also
3842 % makes this look like a page by setting
3843 % \textwidth == \columnwidth == box width
3844 % changes footnotes by redefining:
3845 % \@mpfn == mpfootnote
3846 % \thempfn == \thempfootnote

```

```

3847 %      \@footnotetext == \@mpfootnotetext
3848 %      resets the following list environment parameters
3849 %      \@listdepth == \@mplistdepth
3850 %      where \@mplistdepth is initialized to zero,
3851 %      and executes \@minipagerestore to allow the document
3852 %      style to reset any other parameters it desires.
3853 %      It sets @minipage := T, and resets \everypar to set
3854 %      it false. This switch keeps \addvspace from putting space
3855 %      at the top of a minipage.
3856 %
3857 %      Change added 24 May 89: \minipage sets @minipage globally;
3858 %      \endminipage resets it false.
3859 %
3860 %
3861 % \rule [RAISED]{WIDTH}{HEIGHT} : Makes a WIDTH X HEIGHT rule, raised
3862 %      RAISED.
3863 %
3864 % \underline {TEXT} : Makes an underlined hbox with TEXT in it.
3865 %
3866 % \raisebox{DISTANCE}[HEIGHT][DEPTH]{BOX} : Raises BOX up by DISTANCE
3867 %      length (down if DISTANCE negative). Makes TeX think that
3868 %      the new box extends HEIGHT above the line and DEPTH below, for
3869 %      a total vertical length of HEIGHT+DEPTH. Default values of
3870 %      HEIGHT & DEPTH = actual height and depth of box in new position.
3871 %
3872 % \makebox ==
3873 % BEGIN
3874 %   if next char = (
3875 %     then \@makepicbox
3876 %     else if next char = [
3877 %       then \@makebox
3878 %       else \mbox      fi
3879 %   fi
3880 % END
3881 %
3882 % \@makebox[LEN] ==
3883 % BEGIN
3884 %   leave vertical mode
3885 %   if next char '[' then \@imakebox[LEN]
3886 %     else \@imakebox[LEN][x] fi
3887 % END
3888 %
3889 % \@imakebox[LEN][POS]{OBJ} ==
3890 % BEGIN
3891 %   \hbox to LEN
3892 %     { \mb@l :=L \mb@r :=L \hss
3893 %       \let\mb@POS = \relax
3894 %       \mb@l OBJ \mb@r }
3895 % END
3896 %
3897 % \@makepicbox(X,Y) ==
3898 % BEGIN
3899 %   leave vertical mode
3900 %   if next char = [ then \@imakepicbox(X,Y)
3901 %     else \@imakepicbox(X,Y)[] fi
3902 % END
3903 %
3904 % \@imakepicbox(X,Y)[POS]{OBJ} ==
3905 % BEGIN
3906 %   \vbox to Y * \unitlength
3907 %     { \mb@l :=L \mb@r :=L \hss
3908 %       \mb@t :=L \mb@b :=L \hss
3909 %       tfor \@tempa := POS % one iteration for each token in POS
3910 %         do \mb@eval(\@tempa) :=L null od
3911 %       \mb@t
3912 %       \hbox to X * \unitlength
3913 %         { \mb@l OBJ \mb@r }
3914 %       \mb@b}
3915 % END
3916 %
3917 %
3918 % \def\makebox{\@ifnextchar ({\@makepicbox}{\@ifnextchar
3919 %   [{\@makebox}{\mbox}}}}
3920 %

```



```

3921 \def\mbox#1{\leavevmode\hbox{#1}}
3922
3923 \def\@makebox[#1]{\leavevmode\@ifnextchar [{\@makebox[#1]}]{\@makebox[#1][x]}}
3924
3925 \long\def\@makebox[#1][#2]#3{\hbox to#1{\let\mb@l\hss
3926 \let\mb@r\hss \expandafter\let\csname mb@#2\endcsname\relax
3927 \mb@l #3\mb@r}}
3928
3929 \def\@makepicbox(#1,#2){\leavevmode\@ifnextchar
3930   [{\@makepicbox(#1,#2)}]{\@makepicbox(#1,#2)[]}}
3931
3932 \long\def\@makepicbox(#1,#2)[#3]#4{\vbox to#2\unitlength
3933   {\let\mb@b\vss \let\mb@l\hss\let\mb@r\hss
3934     \let\mb@t\vss
3935     \@tfor\@tempa :=#3\do{\expandafter\let
3936       \csname mb@\@tempa\endcsname\relax}%
3937 \mb@t\hbox to #1\unitlength{\mb@l #4\mb@r}\mb@b}}
3938
3939 \def\newsavebox#1{\@ifdefinable#1{\newbox#1}}
3940
3941 \def\savebox#1{\@ifnextchar ({\@savepicbox#1}{\@ifnextchar
3942   [{\@savebox#1}]{\sbox#1}})}
3943
3944 \def\sbox#1#2{\setbox#1\hbox{#2}}
3945
3946 \def\@savebox#1[#2]{\@ifnextchar [{\@isavebox#1[#2]}]{\@isavebox#1[#2][x]}}
3947
3948 \long\def\@isavebox#1[#2][#3]#4{\setbox#1 \hbox{\@makebox[#2][#3]{#4}}}
3949
3950 \def\@savepicbox#1(#2,#3){\@ifnextchar
3951   [{\@isavepicbox#1(#2,#3)}]{\@isavepicbox#1(#2,#3)[]}}
3952
3953 \long\def\@isavepicbox#1(#2,#3)[#4]#5{\setbox#1 \hbox{\@makepicbox
3954   (#2,#3)[#4]{#5}}}
3955
3956 \def\usebox#1{\leavevmode\copy #1\relax}
3957
3958 %% The following definition of \frame was written by Pavel Curtis
3959 %% (Extra space removed 14 Jan 88)
3960 \long\def\frame#1{\leavevmode
3961   \hbox{\hskip-\@wholewidth
3962     \vbox{\vskip-\@wholewidth
3963       \hrule \@height\@wholewidth
3964         \hbox{\vrule \@width\@wholewidth #1\vrule \@width\@wholewidth}\hrule
3965           \@height \@wholewidth\vskip -\@halfwidth}\hskip-\@wholewidth}}
3966
3967 \newdimen\fbboxrule
3968 \newdimen\fbboxsep
3969
3970 %% (Extra space removed 21 Jun 1991)
3971 \long\def\fbbox#1{\leavevmode\setbox\@tempboxa\hbox{#1}\@tempdima\fbboxrule
3972   \advance\@tempdima \fbboxsep \advance\@tempdima \dp\@tempboxa
3973   \hbox{\lower \@tempdima\hbox
3974     {\vbox{\hrule \@height \fbboxrule
3975       \hbox{\vrule \@width \fbboxrule \hskip\fbboxsep
3976         \vbox{\vskip\fbboxsep \box\@tempboxa\vskip\fbboxsep}\hskip
3977           \fbboxsep\vrule \@width \fbboxrule}%
3978         \hrule \@height \fbboxrule}}}}
3979
3980 \def\framebox{\@ifnextchar ({\@framepicbox}{\@ifnextchar
3981   [{\@framebox}]{\fbbox}})}
3982
3983 \def\@framebox[#1]{\@ifnextchar [{\@iframebox[#1]}]{\@iframebox[#1][x]}}
3984
3985 %% (Extra space removed 21 Jun 1991)
3986 \long\def\@iframebox[#1][#2]#3{\leavevmode
3987   \savebox\@tempboxa[#1][#2]{\kern\fbboxsep #3\kern\fbboxsep}\@tempdima\fbboxrule
3988   \advance\@tempdima \fbboxsep \advance\@tempdima \dp\@tempboxa
3989   \hbox{\lower \@tempdima\hbox
3990     {\vbox{\hrule \@height \fbboxrule
3991       \hbox{\vrule \@width \fbboxrule \hskip-\fbboxrule
3992         \vbox{\vskip\fbboxsep \box\@tempboxa\vskip\fbboxsep}\hskip
3993           -\fbboxrule\vrule \@width \fbboxrule}%
3994         \hrule \@height \fbboxrule}}}}

```

```

3995
3996 \def\@framepicbox(#1,#2){\@ifnextchar
3997   [{\@ifframepicbox(#1,#2)}{\@ifframepicbox(#1,#2) []}]
3998
3999 \long\def\@ifframepicbox(#1,#2)[#3]#4{\frame{\@imakepicbox(#1,#2)[#3]{#4}}}
4000
4001 \def\parbox{\@ifnextchar [{\@iparbox}]{\@iparbox[c]}}
4002
4003 \long\def\@iparbox[#1]#2#3{\leavevmode \@pboxswfalse
4004   \if #1b\vbox
4005     \else \if #1t\vtop
4006       \else \ifmmode \vcenter
4007         \else \@pboxswtrue $\vcenter
4008           \fi
4009     \fi
4010   %% RmS 91/11/04 added \m@th
4011   \fi{\hspace #2\@parboxrestore #3}\if@pboxsw \m@th$\fi}
4012
4013 \let\@dischyph=-
4014 \let\@acci='
4015 \let\@accii='
4016 \let\@acciii=\=
4017
4018
4019 \def\@arrayparboxrestore{\let\par\@par
4020   \let-\@dischyph
4021   \let'\@acci \let'\@accii \let\=\@acciii
4022   \parindent\z@ \parskip\z@
4023   \everypar{\linewidth\hspace
4024     \@totalleftmargin\z@ \leftskip\z@ \rightskip\z@ \@rightskip\z@
4025     \parfillskip\@flushglue \lineskip\normallineskip
4026     \baselineskip\normalbaselineskip\sloppy}
4027
4028 \def\@parboxrestore{\@arrayparboxrestore\let\=\@normalcr}
4029
4030 \newif\if@minipage \@minipagefalse
4031
4032 \def\minipage{\@ifnextchar [{\@iminipage}]{\@iminipage[c]}}
4033
4034 \def\@iminipage[#1]#2{\leavevmode \@pboxswfalse
4035   \if #1b\vbox
4036     \else \if #1t\vtop
4037       \else \ifmmode \vcenter
4038         \else \@pboxswtrue $\vcenter
4039       \fi
4040     \fi
4041     \fi\bgroup
4042     \hspace #2\textwidth\hspace \columnwidth\hspace
4043     \@parboxrestore
4044     \def\@mpfn{\mpfootnote}\def\thempfn{\thempfootnote}\c@mpfootnote\z@
4045     \let\@footnotetext\mpfootnotetext
4046     \let\@listdepth\@mplistdepth \@mplistdepth\z@
4047     \@minipagerestore\global\@minipagetrue %% \global added 24 May 89
4048     \everypar{\global\@minipagefalse\everypar{}}}
4049
4050
4051 \let\@minipagerestore=\relax
4052
4053 \def\endminipage{\par\vskip-\lastskip
4054   \ifvoid\@mpfootins\else
4055     \vskip\skip\@mpfootins\footnoterule\unvbox\@mpfootins\fi
4056   \global\@minipagefalse %% added 24 May 89
4057   \egroup\if@pboxsw \m@th$\fi} %% RmS 91/11/04 added \m@th
4058
4059 \newcount\@mplistdepth
4060 \newinsert\@mpfootins
4061
4062 %% RmS 91/09/29: added \reset@font
4063 \long\def\@mpfootnotetext#1{\global\setbox\@mpfootins
4064   \vbox{\unvbox\@mpfootins
4065     \reset@font\footnotesize
4066     \hspace\columnwidth \@parboxrestore
4067     \edef\@currentlabel{\csname p@mpfootnote\endcsname\@thefnmark}\@makefntext
4068     {\rule{\z@}{\footnotesep}\ignorespaces #1\strut}}}

```

```

4069      % \strut added 27 Mar 89, on suggestion by Don Hosek
4070
4071 \newif\if@pboxsw
4072
4073 \def\rule{\@ifnextchar[{\@rule}{\@rule[\z@]}}
4074
4075 \def\@rule[#1]#2#3{\@tempdima#3\advance\@tempdima #1\leavevmode\hbox{\vrule
4076   \@width#2 \@height\@tempdima \@depth-#1}}
4077
4078 \let\@underline\underline
4079 \def\underline#1{\relax\ifmmode
4080   \@underline{#1}\else $\@underline{\hbox{#1}}\m@th$\relax\fi}
4081
4082 \def\raisebox#1{\@ifnextchar[{\@argsbox{#1}}{\@rsbox{#1}}}]
4083
4084 \def\@argsbox#1[#2]{%
4085 \@ifnextchar[{\@iirsbox{#1}[#2]}{\@irsbox{#1}[#2]}}
4086
4087 \long\def\@rsbox#1#2{\leavevmode\hbox{\raise #1\hbox{#2}}}
4088
4089 \long\def\@irsbox#1[#2]#3{\setbox\@tempboxa \hbox
4090   {\raise #1\hbox{#3}}\ht\@tempboxa#2\leavevmode\box\@tempboxa}
4091
4092 \long\def\@iirsbox#1[#2]#3#4{\setbox\@tempboxa \hbox
4093   {\raise #1\hbox{#4}}\ht\@tempboxa#2\dp\@tempboxa#3\leavevmode\box\@tempboxa}
4094
4095
4096 \message{tabbing,}
4097 %
4098 %   *****
4099 %   *   THE TABBING ENVIRONMENT   *
4100 %   *****
4101 % \dimen{\@firsttab + i} = distance of tab stop i from left margin
4102 %   0 <= i <= 15 (?).
4103 %
4104 % \dimen\@firsttab is initialized to \@totalleftmargin, so it starts
4105 %   at the prevailing left margin.
4106 %
4107 % \maxtab      = number of highest defined tab register
4108 %               probably = \@firsttab + 12
4109 % \nattabmar = tab stop number of next line's left margin
4110 % \curtabmar = tab stop number of current line's left margin
4111 % \curtab    = number of the current tab. At start of line,
4112 %               it equals \curtabmar
4113 % \hightab   = largest tab number currently defined.
4114 % \tabpush   = depth of \pushtab's
4115 %
4116 % \box\@curline = contents of current line, excluding left margin skip,
4117 %               and excluding contents of current field
4118 % \box\@curfield = contents of current field
4119 %
4120 % @rjfield     = switch: T iff the last field of the line should be
4121 %               right-justified at the right margin.
4122 %
4123 % \tabbingsep = distance left by the \ command between the current
4124 %               position and the field that is 'left-shifted'.
4125 %
4126 % UTILITY MACROS
4127 % \@stopfield : closes the current field
4128 % \@addfield  : adds the current field to the current line.
4129 % \@contfield : continues the current field
4130 % \@startfield : begins the next field
4131 % \@stopline  : closes the current line and outputs it
4132 % \@startline : starts the next line
4133 % \@ifatmargin : an \if that is true iff the current line.
4134 %               has width zero
4135 %
4136 % \@startline ==
4137 % BEGIN
4138 % \@curtabmar :=G \@nattabmar
4139 % \@curtab :=G \@curtabmar
4140 % \box\@curline :=G null
4141 % \@startfield
4142 % \strut

```

```

4143 % END
4144 %
4145 % \@stopline ==
4146 % BEGIN
4147 % \unskip
4148 % \@stopfield
4149 % if @rjfield = T
4150 % then @rjfield :=G F
4151 % \@tempdima := \@totalleftmargin + \linewidth
4152 % \hbox to \@tempdima{\@itemfudge
4153 % \hskip \dimen\@curtabmar
4154 % \box\@curline
4155 % \hfil
4156 % \box\@curfield}
4157 % else \@addfield
4158 % \hbox {\@itemfudge
4159 % \hskip \dimen\@curtabmar
4160 % \box\@curline}
4161 % fi
4162 % END
4163 %
4164 % \@startfield ==
4165 % BEGIN
4166 % \box\@curfield :=G \hbox {
4167 % END
4168 %
4169 % \@stopfield ==
4170 % BEGIN
4171 % }
4172 % END
4173 %
4174 % \@contfield ==
4175 % BEGIN
4176 % \box\@curfield :=G \hbox { \unhbox\@currfield %%} brace matching
4177 % END
4178 % \@addfield ==
4179 % BEGIN
4180 % \box\@curline :=G \unbox\@curline * \unbox\@curfield
4181 % END
4182 %
4183 % \@ifatmargin ==
4184 % BEGIN
4185 % if dim of \box\@curline = Opt then
4186 % END
4187 %
4188 %
4189 % \tabbing ==
4190 % BEGIN
4191 % \lineskip :=L Opt
4192 % \> == \@rtab
4193 % \< == \@ltab
4194 % \= == \@settab
4195 % \+ == \@tabplus
4196 % \- == \@tabminus
4197 % \' == \@tabrj
4198 % \' == \@tablab
4199 % \\\ == BEGIN \@stopline \@startline END
4200 % \|[DIST] == BEGIN \@stopline \vskip DIST \@startline\ignorespaces END
4201 % \|* == BEGIN \@stopline \penalty 10000 \@startline END
4202 % \|[DIST] == BEGIN \@stopline \penalty 10000 \vskip DIST
4203 % \@startline\ignorespaces END
4204 % \@righttab :=G \@nattabmar :=G \@firsttab
4205 % \@tabpush :=G 0
4206 % \dimen\@firsttab := \@totalleftmargin
4207 % @rjfield :=G F
4208 % \trivlist \item[]
4209 % if @minipage = F then \vskip \parskip fi
4210 % \box\@tabfbox = \rlap{\indent\the\everypar} % note: \the\everypar sets
4211 % \@itemfudge == BEGIN \box\@tabfbox END % @inlabel :=G F
4212 % \@startline
4213 % \ignorespaces
4214 % END
4215 %
4216 % \@endtabbing ==

```

```

4217 % BEGIN
4218 % \@stopline
4219 % if \@tabpush > 0 then error message: 'unmatched \poptabs' fi
4220 % \endtrivlist
4221 % END
4222 %
4223 % \@rtab ==
4224 % BEGIN
4225 % \@stopfield
4226 % \@addfield
4227 % if \@curtab < \@hightab
4228 % then \@curtab :=G \@curtab + 1
4229 % else error message 'Undefined Tab' fi
4230 % \@tempdima := \dimen\@curtab - \dimen\@curtabmar
4231 % - width of box \@curline
4232 % \box\@curline :=G \hbox{\unhbox\@curline + \hskip\@tempdima}
4233 % \@startfield
4234 % END
4235 %
4236 % \@settab ==
4237 % BEGIN
4238 % \@stopfield
4239 % \@addfield
4240 % if \@curtab < \@maxtab
4241 % then \@curtab :=G \@curtab+1
4242 % else error message: 'Too many tabs' fi
4243 % if \@curtab > \@hightab
4244 % then \@hightab :=L \@curtab fi
4245 % \dimen\@curtab :=L \dimen\@curtabmar + width of \box\@curline
4246 % \@startfield
4247 % END
4248 %
4249 % \@ltab ==
4250 % BEGIN
4251 % \@ifatmargin
4252 % then if \@curtabmar > \@firsttab
4253 % then \@curtab :=G \@curtab - 1
4254 % \@curtabmar :=G \@curtabmar - 1
4255 % else error message 'Too many untab' fi
4256 % else error message 'Left tab in middle of line'
4257 % fi
4258 % END
4259 %
4260 % \@tabplus ==
4261 % BEGIN
4262 % if \@nxttabmar < \@hightab
4263 % then \@nxttabmar :=G \@nxttabmar+1
4264 % else error message 'Undefined tab'
4265 % fi
4266 % END
4267 %
4268 % \@tabminus ==
4269 % BEGIN
4270 % if \@nxttabmar > \@firsttab
4271 % then \@nxttabmar :=G \@nxttabmar-1
4272 % else error message 'Too many untab'
4273 % fi
4274 % END
4275 %
4276 % \@tabrj ==
4277 % BEGIN \@stopfield
4278 % \@addfield
4279 % @rjfield :=G T
4280 % \@startfield
4281 % END
4282 %
4283 % \@tablab ==
4284 % BEGIN \@stopfield
4285 % \box\@curline G:= \hbox{\box\@curline %% 'G' added 17 Jun 86
4286 % \hskip - width of \box\@curfield
4287 % \hskip -\tabbingsep
4288 % \box\@curfield
4289 % \hskip \tabbingsep }
4290 % \@startfield

```

```

4291 % END
4292 %
4293 % \pushtabs ==
4294 % BEGIN
4295 %   \@stopfield
4296 %   \@tabpush :=G \@tabpush + 1
4297 %   \beginngroup
4298 %   \@contfield
4299 % END
4300 %
4301 % \poptabs ==
4302 % BEGIN
4303 %   \@stopfield
4304 %   if \@tabpush > 0
4305 %     then \endgroup
4306 %         \@tabpush :=G \@tabpush - 1
4307 %     else error message: ‘Too many \poptabs’
4308 %   fi
4309 %   \@contfield
4310 % END
4311 %
4312 % The accents \‘, \’ , and \= that have been redefined inside a tabbing
4313 % environment can be called by typing \a‘, \a’ , and \a=.
4314 %
4315
4316 \expandafter \let \csname a‘\endcsname = \‘
4317 \expandafter \let \csname a’\endcsname = \’
4318 \expandafter \let \csname a=\endcsname = \=
4319 \def\a#1{\csname a#1\endcsname}
4320
4321 \newif\if@rjfield
4322 \newcount\@firsttab
4323 \newcount\@maxtab
4324 \newdimen\@gtempa \@firsttab=\allocationnumber
4325 \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa
4326 \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa
4327 \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa \newdimen\@gtempa
4328 \newdimen\@gtempa \@maxtab=\allocationnumber
4329 \dimen\@firsttab=0pt
4330 \newcount\@nxttabmar
4331 \newcount\@curtabmar
4332 \newcount\@curtab
4333 \newcount\@hightab
4334 \newcount\@tabpush
4335 \newbox\@curline
4336 \newbox\@curfield
4337 \newbox\@tabfbox
4338
4339 \def\@startline{\global\@curtabmar\@nxttabmar
4340   \global\@curtab\@curtabmar\global\setbox\@curline\hbox % missing \global
4341   }\@startfield\strut} % added 17 Jun 86
4342
4343 \def\@stopline{\unskip\@stopfield\if@rjfield \global\@rjfieldfalse
4344   \@tempdima\@totalleftmargin \advance\@tempdima\linewidth
4345   \hbox to\@tempdima{\@itemfudge\hskip\dimen\@curtabmar
4346     \box\@curline\hfil\box\@curfield}\else\@addfield
4347     \hbox{\@itemfudge\hskip\dimen\@curtabmar\box\@curline}\fi}
4348
4349 \def\@startfield{\global\setbox\@curfield\hbox\bgroup}%{ BRACE MATCH HACK
4350   \let\@stopfield=}
4351 \def\@contfield{\global\setbox\@curfield\hbox\bgroup\unhbox\@curfield}
4352 \def\@addfield{\global\setbox\@curline\hbox{\unhbox
4353   \@curline\unhbox\@curfield}}
4354 \def\@ifatmargin{\ifdim \wd\@curline =\z@}
4355
4356 \def\@tabcr{\@stopline \@ifstar{\penalty \@M \@xtabcr}{\@xtabcr}}
4357
4358 \def\@xtabcr{\@ifnextchar[{\@itabcr}{\@startline\ignorespaces}}
4359
4360 \def\@itabcr[#1]{\vskip #1\@startline\ignorespaces}
4361
4362 \def\kill{\@stopfield\@startline\ignorespaces}
4363
4364 % REMOVE \outer FROM PLAIN’S DEF OF \+

```

```

4365
4366 \def\+{\tabalign}
4367
4368
4369 \def\tabbing{\lineskip \z@\let\>\@rtab\let\<\@ltab\let\=@\settab
4370 \let\+@\tabplus\let\-\@tabminus\let\'\@tabrj\let\'\@tablab
4371 \let\=\@tabcr
4372 \global\@hightab\@firsttab
4373 \global\@nxttabmar\@firsttab
4374 \dimen\@firsttab\@totalleftmargin
4375 \global\@tabpush\z@ \global\@rjfieldfalse
4376 \trivlist \item[]\if@minipage\else\vskip\parskip\fi
4377 \setbox\@tabfbox\hbox{\rlap{\indent\hskip\@totalleftmargin
4378 \the\everypar}}}\def\@itemfudge{\box\@tabfbox}\@startline\ignorespaces}
4379
4380 \def\endtabbing{\@stopline\ifnum\@tabpush >\z@ \@badpoptabs \fi\endtrivlist}
4381
4382 \def\@rtab{\@stopfield\@addfield\ifnum \@curtab <\@hightab
4383 \global\advance\@curtab \@one \else\@badtab\fi
4384 \@tempdima\dimen\@curtab
4385 \advance\@tempdima -\dimen\@curtabmar
4386 \advance\@tempdima -\wd\@curline
4387 \global\setbox\@curline\hbox{\unhbox\@curline\hskip\@tempdima}}%
4388 \@startfield\ignorespaces}
4389 % Omitted \global added to \@rtab 17 Jun 86
4390
4391 \def\@settab{\@stopfield\@addfield\ifnum \@curtab <\@maxtab
4392 \global\advance\@curtab \@one \else\@latexerr{Tab overflow}\@ehd\fi
4393 \ifnum\@curtab >\@hightab
4394 \@hightab\@curtab\fi
4395 \dimen\@curtab\dimen\@curtabmar
4396 \advance\dimen\@curtab \wd\@curline\@startfield\ignorespaces}
4397 \def\@ltab{\@ifatmargin\ifnum\@curtabmar >\@firsttab
4398 \global\advance\@curtab \m@ne \global\advance\@curtabmar \m@ne \else
4399 \@badtab\fi\else
4400 \@latexerr{\string\<\space in mid line}\@ehd\fi\ignorespaces}
4401 \def\@tabplus {\ifnum \@nxttabmar <\@hightab
4402 \global\advance\@nxttabmar \@one \else
4403 \@badtab\fi\ignorespaces}
4404 \def\@tabminus{\ifnum\@nxttabmar >\@firsttab
4405 \global\advance\@nxttabmar \m@ne \else
4406 \@badtab\fi\ignorespaces}
4407 \def\@tabrj{\@stopfield\@addfield\global\@rjfieldtrue\@startfield\ignorespaces}
4408
4409 \def\@tablab{\@stopfield\global\setbox\@curline\hbox{\box\@curline
4410 \hskip -\wd\@curfield \hskip -\tabbingsep \box\@curfield
4411 \hskip \tabbingsep}\@startfield\ignorespaces}
4412 % \setbox\@curline made \global in \@tablab. 17 Jun 86
4413
4414 \def\pushtabs{\@stopfield\@addfield\global\advance\@tabpush \@one \begingroup
4415 \@contfield}
4416 \def\poptabs{\@stopfield\@addfield\ifnum\@tabpush >\z@ \endgroup
4417 \global\advance\@tabpush \m@ne \else
4418 \@badpoptabs\fi\@contfield}
4419
4420 \newdimen\tabbingsep
4421
4422 \message{array,}
4423 % *****
4424 % * ARRAY AND TABULAR ENVIRONMENTS *
4425 % *****
4426 %
4427 % ARRAY PARAMETERS:
4428 % \arraycolsep : half the width separating columns in an array environment
4429 % \tabcolsep : half the width separating columns in a tabular environment
4430 % \arrayrulewidth : width of rules
4431 % \doublerulesep : space between adjacent rules in array or tabular
4432 % \arraystretch : line spacing in array and tabular environments is done by
4433 % placing a strut in every row of height and depth
4434 % \arraystretch times the height and depth of the strut
4435 % produced by an ordinary \strut command.
4436 %
4437 % PREAMBLE:
4438 % The PREAMBLE argument of an array or tabular environment can contain

```

```

4439 % the following:
4440 %   l,r,c : indicate where entry is to be placed.
4441 %   |      : for vertical rule
4442 %   @{EXP} : inserts the text EXP in every column. \arraycolsep or \tabcolsep
4443 %           spacing is suppressed.
4444 %   *{N}{PRE} : equivalent to writing N copies of PRE in the preamble. PRE
4445 %             may contain *{N'}{EXP'} expressions.
4446 %   p{LEN} : makes entry in parbox of width LEN.
4447 %
4448 % SPECIAL ARRAY COMMANDS:
4449 %   \multicolumn{N}{FORMAT}{ITEM} : replaces the next N column items by
4450 %   ITEM, formatted according to FORMAT. FORMAT should contain at most
4451 %   one l,r or c. If it contains none, then ITEM is ignored.
4452 %
4453 %   \vline : draws a vertical line the height of the current row. May
4454 %           appear in an array element entry.
4455 %   \hline : draws a horizontal line between rows. Must appear either
4456 %           before the first entry (to appear above the first row) or right
4457 %           after a \\ command. If followed by another \hline, then adds
4458 %           a \vskip of \doublerulesep.
4459 %
4460 %   \cline[i-j] : draws horizontal lines between rows covering columns
4461 %               i through j, inclusive. Multiple commands may follow
4462 %               one another to provide lines covering several disjoint
4463 %               columns
4464 %   \extracolsep{WIDTH} : for use inside an @ in the preamble. Causes a WIDTH
4465 %                       space to be added between columns for the rest of the
4466 %                       columns. This is in addition to the ordinary intercolumn
4467 %                       space.
4468 %
4469 % \array ==
4470 % BEGIN
4471 %   \@acol == \@arrayacol
4472 %   \@classz == \@arrayclassz
4473 %   \@classiv == \@arrayclassiv
4474 %   \\ == \@arraycr
4475 %   \@halignto == NULL
4476 %   \@tabarray
4477 % END
4478 %
4479 % \endarray{NAME} == BEGIN \cr cr }} END
4480 %
4481 % \tabular ==
4482 % BEGIN
4483 %   \@halignto == NULL
4484 %   \@tabular
4485 % END
4486 %
4487 % \tabular*{WIDTH} ==
4488 % BEGIN
4489 %   \@halignto == to WIDTH
4490 %   \@tabular
4491 % END
4492 %
4493 % \@tabular ==
4494 % BEGIN
4495 %   \leavevmode
4496 %   \hbox { $
4497 %     \@acol == \@tabacol
4498 %     \@classz == \@tabclassz
4499 %     \@classiv == \@tabclassiv
4500 %     \\ == \@tabularcr
4501 %     \@tabarray
4502 %   }
4503 % END
4504 % \endtabular == BEGIN \cr cr}} $} END
4505 %
4506 % \@tabarray == if next char = [ then \@array else \@array[c] fi
4507 %
4508 % \@array[POS]{PREAMBLE} ==
4509 % BEGIN
4510 %   define \@arstrutbox to make \@arstrut produce strut of height
4511 %   and depth \arraystretch times the height and
4512 %   depth of a normal strut.

```



```

4513 % \mkpream{PREAMBLE}
4514 % \@preamble == \halign \@halignto {\tabskip=Opt\@arstrut
4515 % eval{\@preamble}\tabskip = Opt\cr %% }
4516 % \@startpbox == \@startpbox
4517 % \@endpbox == \@endpbox
4518 % if POS = t then \vtop
4519 % else if POS = b then \vbox
4520 % else \vcenter
4521 % fi fi
4522 % {
4523 % \par ==L \relax
4524 % \sharp == #
4525 % \protect == \relax
4526 % \lineskip :=L Opt
4527 % \baselineskip :=L Opt
4528 % \@preamble
4529 % END
4530 %
4531 %
4532 % \@arraycr ==
4533 % BEGIN
4534 % $ %% Prevents extra space at end of row's last entry.
4535 % if next char = [
4536 % then \@argarraycr
4537 % else $ \cr %% Needed to balance $
4538 % END
4539 %
4540 % \@argarraycr[LENGTH] ==
4541 % BEGIN
4542 % $ %% Needed to balance $ of \@arraycr
4543 % if LENGTH > 0
4544 % then \@tempdima := depth of \@arstrutbox + LENGTH
4545 % \vrule height Opt width Opt depth \@tempdima
4546 % \cr
4547 % else \cr \noalign{\vskip LENGTH}
4548 % END
4549 %
4550 % \@tabularcr and \@argtabularcr same as \@arraycr and \@argarraycr
4551 % except without the extra $'s.
4552 %
4553 \def\extracolsep#1{\tabskip #1\relax}
4554 %
4555 \def\array{\let\@acol\@arrayacol \let\@classz\@arrayclassz
4556 \let\@classiv\@arrayclassiv \let\@arraycr\let\@halignto\@empty\@tabarray}
4557 %
4558 \def\endarray{\crr\egroup\egroup}
4559 \def\endtabular{\crr\egroup\egroup $\egroup}
4560 \expandafter \let \csname endtabular*\endcsname = \endtabular
4561 %
4562 \def\tabular{\let\@halignto\@empty\@tabular}
4563 %
4564 \expandafter \def\csname tabular*\endcsname #1{\def\@halignto{to#1}\@tabular}
4565 %
4566 \def\@tabular{\leavevmode \hbox \bgroup $\let\@acol\@tabacol
4567 \let\@classz\@tabclassz
4568 \let\@classiv\@tabclassiv \let\@tabularcr\@tabarray}
4569 %
4570 %% RmS 91/11/04 added \m@th
4571 \def\@tabarray{\m@th\@ifnextchar[{\@array}{\array[c]}}
4572 %
4573 \def\@array[#1]#2{\setbox\@arstrutbox\hbox{\vrule
4574 \@height\arraystretch \ht\strutbox
4575 \@depth\arraystretch \dp\strutbox
4576 \@widthz@}\mkpream{#2}\edef\@preamble{\halign \noexpand\@halignto
4577 \bgroup \tabskipz@ \@arstrut \@preamble \tabskipz@ \cr}%
4578 \let\@startpbox\@startpbox \let\@endpbox\@endpbox
4579 \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi\fi
4580 \bgroup \let\par\relax
4581 \let\@sharp#\let\protect\relax \lineskipz@\baselineskipz@\@preamble}
4582 %
4583 \def\@arraycr{${\ifnum0=} \fi\@ifstar{\@arraycr}{\@xarraycr}}
4584 \def\@xarraycr{\@ifnextchar[{\@argarraycr}{\ifnum0={\fi}$}\cr}}
4585 %
4586 \def\@argarraycr[#1]{\ifnum0={\fi}$}\ifdim #1>z@ \@xargarraycr[#1]\else

```

```

4587 \yargarraycr{#1}\fi}
4588
4589 \def\@tabularcr{\ifnum0='}\fi\@ifstar{\@tabularcr}{\@tabularcr}}
4590 \def\@xtabularcr{\@ifnextchar[{\@argtabularcr}{\ifnum0='}\fi}\cr}}
4591
4592 \def\@argtabularcr[#1]{\ifnum0='}\fi\ifdim #1>\z@
4593 \unskip\@xargarraycr{#1}\else \@yargarraycr{#1}\fi}
4594
4595 \def\@xargarraycr#1{\@tempdima #1\advance\@tempdima \dp \@arstrutbox
4596 \vrule \@height\z@ \@depth\@tempdima \@width\z@ \cr}
4597
4598 \def\@yargarraycr#1{\cr\noalign{\vskip #1}}
4599
4600
4601 % \multicolumn{NUMBER}{FORMAT}{ITEM} ==
4602 % BEGIN
4603 % \multispan{NUMBER}
4604 % \begingroup
4605 % \@addamp == null
4606 % \@mkpream{FORMAT}
4607 % \@sharp == ITEM
4608 % \protect == \relax
4609 % \@startpbox == \@startpbox
4610 % \@endpbox == \@endpbox
4611 % \@arstrut
4612 % \@preamble
4613 % \endgroup
4614 % END
4615
4616 % The command \def\@addamp{} was removed from \multicolumn on 6 Dec 86
4617 % because it caused embedded array environments not to work. I think
4618 % that it was included originally to prevent an error message if
4619 % the 2nd argument to the \multicolumn command had two column specifiers.
4620 %
4621 % 8 Feb 89 - \hbox{} added after \@preamble to correct bug that
4622 % occurred if \multicolumn preceded \[D] with D > 0,
4623 % caused by \[[] command doing an \unskip, which removed
4624 % \tabcolsep glue inserted by \multicolumn
4625
4626 \def\multicolumn#1#2#3{\multispan{#1}\begingroup
4627 \@mkpream{#2}%
4628 \def\@sharp{#3}\let\protect\relax
4629 \let\@startpbox\@startpbox\let\@endpbox\@endpbox
4630 \@arstrut \@preamble\hbox{}\endgroup\ignorespaces}
4631
4632
4633 % Codes for classes and character numbers of array, tabular and
4634 % multicolumn arguments.
4635 %
4636 % Character Class Number
4637 % -----
4638 % c 0 0
4639 % l 0 1
4640 % r 0 2
4641 %
4642 % / 1 -
4643 % @ 2 -
4644 % p 3 -
4645 % {@-exp} 4 -
4646 % {p-arg} 5 -
4647 %
4648 % \@testpach \foo : expands \foo, which should be an array parameter token,
4649 % and sets \@chclass and \@chnum to its class and number.
4650 % Uses \@lastchclass to distinguish 4 and 5
4651 %
4652 % Preamble error codes
4653 % 0: 'illegal character'
4654 % 1: 'Missing @-exp'
4655 % 2: 'Missing p-arg'
4656 %
4657 % \@addamp ==
4658 % BEGIN if @firstamp = true then @firstamp := false
4659 % else & fi
4660 % END

```

```

4661 %
4662 % \mkpream TOKENLIST ==
4663 % BEGIN
4664 %   @firstamp := T
4665 %   \@lastchclass := 6
4666 %   \@preamble == null
4667 %   \@sharp == \relax
4668 %   \protect == BEGIN \noexpand\protect\noexpand END
4669 %   \@startpbox == \relax
4670 %   \@endpbox == \relax
4671 %   \@expast{TOKENLIST}
4672 %   for \@nextchar := expand(\@tempa)
4673 %     do \@testpach{\@nextchar}
4674 %       case of \@chclass
4675 %         0 -> \@classz
4676 %         1 -> \@classi
4677 %         ...
4678 %         5 -> \@classv
4679 %       end case
4680 %       \@lastchclass := \@chclass
4681 %     od
4682 %   case of \@lastchclass
4683 %     0 -> \hskip \arraycolsep % lrc
4684 %     1 -> % /
4685 %     2 -> \@preamerr1 % 'Missing @-exp' % @
4686 %     3 -> \@preamerr2 % 'Missing p-arg' % p
4687 %     4 -> % @-exp
4688 %     5 -> \hskip \arraycolsep % p-exp
4689 %   end case
4690 % END
4691 %
4692 % \@arrayclassz ==
4693 % BEGIN
4694 %   \@preamble := \@preamble *
4695 %   case of \@lastchclass
4696 %     0 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
4697 %     1 -> \@addamp \hskip \arraycolsep
4698 %     2 -> % impossible
4699 %     3 -> % impossible
4700 %     4 -> \@addamp
4701 %     5 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
4702 %     6 -> \@addamp \hskip \arraycolsep
4703 %   end case
4704 %   * case of \@chnum
4705 %     0 -> \hfil$\relax\@sharp$\hfil
4706 %     1 -> $\relax\@sharp$\hfil
4707 %     2 -> \hfil$\relax\@sharp$
4708 %   end case
4709 % END
4710 %
4711 % \@tabclassz == similar to \@arrayclassz
4712 %
4713 % \@classi ==
4714 % BEGIN
4715 %   \@preamble := \@preamble *
4716 %   case of \@lastchclass
4717 %     0 -> \hskip \arraycolsep \@arrayrule
4718 %     1 -> \hskip \doublerulesep \@arrayrule
4719 %     2 -> % impossible
4720 %     3 -> % impossible
4721 %     4 -> \@arrayrule
4722 %     5 -> \hskip \arraycolsep \@arrayrule
4723 %     6 -> \@arrayrule
4724 %   end case
4725 % END
4726 %
4727 % \@classi ==
4728 % BEGIN
4729 %   \@preamble := \@preamble *
4730 %   case of \@lastchclass
4731 %     0 ->
4732 %     1 -> \hskip .5\arrayrulewidth
4733 %     2 -> % impossible
4734 %   else ->

```

```

4735 %                end case
4736 % END
4737 %
4738 % \@classiii ==
4739 % BEGIN
4740 %   \@preamble := \@preamble *
4741 %           case of \@lastchclass
4742 %             0 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
4743 %             1 -> \@addamp \hskip \arraycolsep
4744 %             2 -> % impossible
4745 %             3 -> % impossible
4746 %             4 -> \@addamp
4747 %             5 -> \hskip \arraycolsep \@addamp \hskip \arraycolsep
4748 %             6 -> \@addamp \hskip \arraycolsep
4749 %           end case
4750 % END
4751 %
4752 % \@arrayclassiv == BEGIN \@preamble := \@preamble * $ \@nextchar$ END
4753 %
4754 % \@tabclassiv == same as \@arrayclassv except without the $ ... $
4755 %
4756 % \@classv ==
4757 % BEGIN
4758 %   \@preamble := \@preamble * \@startpbox{\@nextchar}\ignorespaces\sharp
4759 %               \@endpbox
4760 % END
4761 %
4762 % \@expast{S}: Sets \@tempa := S with all instances of *{N}{STRING}
4763 %             replaced by N copies of STRING, where N > 0. An *
4764 %             appearing inside braces is ignored, but *-expressions
4765 %             inside STRING are expanded, so nested *-expressions are
4766 %             handled properly.
4767 %
4768 % \@expast{S} == BEGIN \@expast S *0x\@@ END
4769 %
4770 % \@expast S1 *{N}{S2} S3 \@@ ==
4771 % BEGIN
4772 %   \@tempa := S1
4773 %   \@tempcnta := N
4774 %   if \@tempcnta > 0
4775 %     then while \@tempcnta > 0 do \@tempa := \@tempa S2
4776 %                               \@tempcnta := \@tempcnta - 1 od
4777 %     \@tempb == \@expast
4778 %     else \@tempb == \@exnoop
4779 %   fi
4780 %   \expandafter \@tempb \@tempa S3 \@@
4781 % END
4782 %
4783 %
4784 \def\@exnoop #1\@@{}
4785 %
4786 \def\@expast#1{\@expast #1*0x\@@}
4787 %
4788 \def\@expast#1*#2#3#4\@@{\edef\@tempa{#1}\@tempcnta#2\relax
4789   \ifnum\@tempcnta >\z@ \@whilenum\@tempcnta >\z@\do
4790     {\edef\@tempa{\@tempa#3}\advance\@tempcnta \m@ne}\let\@tempb\@expast
4791   \else \let\@tempb\@exnoop\fi
4792   \expandafter\@tempb \@tempa #4\@@}
4793 %
4794 %
4795 \newif\if@firstamp
4796 \def\@addamp{\if@firstamp \@firstampfalse \else
4797   \edef\@preamble{\@preamble &}\fi}
4798 \def\@arrayacol{\edef\@preamble{\@preamble \hskip \arraycolsep}}
4799 \def\@tabacol{\edef\@preamble{\@preamble \hskip \tabcolsep}}
4800 \def\@ampacol{\@addamp \@acol}
4801 \def\@acolampacol{\@acol\@addamp\@acol}
4802 %
4803 \def\@mkpream#1{\@firstamptrue\@lastchclass6
4804 \let\@preamble\@empty\def\protect{\noexpand\protect\noexpand}\let\@sharp\relax
4805 \let\@startpbox\relax\let\@endpbox\relax
4806 \@expast{#1}\expandafter\@tfor \expandafter
4807 \@nextchar \expandafter:\expandafter=\@tempa\do{\@testpach\@nextchar
4808 \ifcase \@chclass \@classz \or \@classi \or \@classii \or \@classiii

```

```

4809     \or \@classiv \or \@classv \fi \@lastchclass \@chclass}%
4810 \ifcase \@lastchclass \@acol
4811     \or \or \@preamerr \@ne \or \@preamerr \tw@ \or \or \@acol \fi}
4812
4813 \def \@arrayclassz{\ifcase \@lastchclass \@acolampacol \or \@ampacol \or
4814     \or \or \@addamp \or
4815     \@acolampacol \or \@firstampfalse \@acol \fi
4816 \edef \@preamble{\@preamble
4817     \ifcase \@chnum
4818     \hfil $\relax \@sharp$ \hfil \or $\relax \@sharp$ \hfil
4819     \or \hfil $\relax \@sharp$ \fi}}
4820
4821 %% RmS 91/08/14 inserted extra braces around entry for NFSS
4822 \def \@tabclassz{\ifcase \@lastchclass \@acolampacol \or \@ampacol \or
4823     \or \or \@addamp \or
4824     \@acolampacol \or \@firstampfalse \@acol \fi
4825 \edef \@preamble{\@preamble{%
4826     \ifcase \@chnum
4827     \hfil \ignorespaces \@sharp \unskip \hfil
4828     \or \ignorespaces \@sharp \unskip \hfil
4829     \or \hfil \hskip \z@ \ignorespaces \@sharp \unskip \fi}}}
4830
4831 \def \@classi{\ifcase \@lastchclass \@acol \@arrayrule \or
4832     \@addtopreamble{\hskip \doublerulesep} \@arrayrule \or
4833     \or \or \@arrayrule \or
4834     \@acol \@arrayrule \or \@arrayrule \fi}
4835
4836
4837 \def \@classii{\ifcase \@lastchclass \or
4838     \@addtopreamble{\hskip .5\arrayrulewidth} \fi}
4839
4840 \def \@classiii{\ifcase \@lastchclass \@acolampacol \or
4841     \@addamp \@acol \or
4842     \or \or \@addamp \or
4843     \@acolampacol \or \@ampacol \fi}
4844
4845 \def \@tabclassiv{\@addtopreamble \@nextchar}
4846
4847 \def \@arrayclassiv{\@addtopreamble {$ \@nextchar $}}
4848
4849 \def \@classv{\@addtopreamble {\@startpbox {\@nextchar} \ignorespaces
4850 \@sharp \@endpbox}}
4851
4852 \def \@addtopreamble#1{\edef \@preamble{\@preamble #1}}
4853
4854 \newcount \@chclass
4855 \newcount \@lastchclass
4856 \newcount \@chnum
4857
4858 \newdimen \arraycolsep
4859 \newdimen \tabcolsep
4860 \newdimen \arrayrulewidth
4861 \newdimen \doublerulesep
4862
4863 \def \arraystretch{1} % Default value.
4864
4865 \newbox \@arstrutbox
4866 \def \@arstrut{\relax \ifmmode \copy \@arstrutbox \else \unhcopy \@arstrutbox \fi}
4867
4868
4869 \def \@arrayrule{\@addtopreamble {\hskip -.5\arrayrulewidth
4870 \vrule \@width \arrayrulewidth \hskip -.5\arrayrulewidth}}
4871
4872 \def \@testpach#1{\@chclass \ifnum \@lastchclass=\tw@ 4 \else
4873 \ifnum \@lastchclass=3 5 \else
4874 \z@ \if #1c \@chnum \z@ \else
4875 \if #1l \@chnum \@ne \else
4876 \if #1r \@chnum \tw@ \else
4877 \@chclass \if #1\ \@ne \else
4878 \if #1@ \tw@ \else
4879 \if #1p3 \else \z@ \@preamerr 0 \fi
4880 \fi \fi \fi \fi \fi \fi \fi
4881 \fi}
4882

```

```

4883 \def\hline{\noalign{\ifnum0='}\fi\hrule \@height \arrayrulewidth \futurelet
4884   \@tempa\@xhline}
4885
4886 \def\@xhline{\ifx\@tempa\hline\vskip \doublerulesep\fi
4887   \ifnum0='{\fi}}
4888
4889 \def\vline{\vrule \@width \arrayrulewidth}
4890
4891 \newcount\@cla
4892 \newcount\@clb
4893
4894 \def\cline#1{\@cline[#1]}
4895 \def\@cline[#1-#2]{\noalign{\global\@cla#1\relax
4896 \global\advance\@cla\m@ne
4897 \ifnum\@cla>\z@\global\let\@gtempa\@clinea\else
4898   \global\let\@gtempa\@clineb\fi
4899 \global\@clb#2\relax
4900 \global\advance\@clb-\@cla}\@gtempa
4901 \noalign{\vskip-\arrayrulewidth}}
4902
4903 \def\@clinea{\multispan\@cla&\multispan\@clb
4904 \unskip\leaders\hrule \@height \arrayrulewidth \hfill
4905 \cr}
4906
4907 \def\@clineb{\multispan\@clb
4908 \unskip\leaders\hrule \@height \arrayrulewidth \hfill
4909 \cr}
4910
4911 % \@startpbox{WIDTH} TEXT \egroup == \parbox{WIDTH}{TEXT}
4912 % \@endpbox == \unskip \strut \par \egroup\hfil (Changed 14 Jan 89)
4913 %
4914
4915 \def\@startpbox#1{\vtop\bgroup \hsize #1\@arrayparboxrestore}
4916 \def\@endpbox{\unskip\strut\par\egroup\hfil}
4917
4918 % 14 Jan 89: Def of \@endpbox changed from
4919 %   \def\@endpbox{\par\vskip\dp\@arstrutbox\egroup\hfil}
4920 % so vertical spacing works out right if the last line of a 'p' entry
4921 % has a descender.
4922
4923 \let\@startpbox=\@startpbox
4924 \let\@endpbox=\@endpbox
4925
4926 \message{picture,}
4927 % *****
4928 % * THE PICTURE ENVIRONMENT *
4929 % *****
4930 %
4931 % \unitlength = value of dimension argument
4932 % \@wholewidth = current line width
4933 % \@halfwidth = half of current line width
4934 % \@linefnt = font for drawing lines
4935 % \@circlefnt = font for drawing circles
4936 %
4937 % \linethickness{DIM} : Sets the width of horizontal and vertical lines
4938 % in a picture to DIM. Does not change width of slanted lines
4939 % or circles. Width of all lines reset by \thinlines and
4940 % \thicklines
4941 %
4942 % \picture(XSIZE,YSIZE)(XORG,YORG)
4943 % BEGIN
4944 %   \@picht :=L YSIZE * \unitlength
4945 %   box \@picbox :=
4946 %     \hbox to XSIZE * \unitlength
4947 %       {\hskip -XORG * \unitlength
4948 %         \lower YORG * \unitlength
4949 %         \hbox{
4950 %           \ignorespaces %% added 13 June 89
4951 %         }
4952 %     }
4953 % \endpicture ==
4954 % BEGIN
4955 %   } \hss }
4956 % height of \@picbox := \@picht

```

```

4957 %           depth of \@picbox := 0
4958 %           \mbox{\box\@picbox} %% change 26 Aug 91
4959 %   END
4960 %
4961 % \put(X, Y){OBJ} ==
4962 %   BEGIN
4963 %     \killglue
4964 %     \raise Y * \unitlength \hbox to Opt { \hskip X * \unitlength
4965 %                                           OBJ \hss
4966 %                                           }
4966 %     \ignorespaces
4967 %   END
4968 %
4969 % \multiput(X, Y) (DELX, DELY){N}{OBJ} ==
4970 %   BEGIN
4971 %     \killglue
4972 %     \@multicnt := N
4973 %     \@xdim := X * \unitlength
4974 %     \@ydim := Y * \unitlength
4975 %     while \@multicnt > 0
4976 %       do \raise \@ydim \hbox to Opt { \hskip \@xdim
4977 %                                       OBJ \hss }
4978 %       \@multicnt := \@multicnt - 1
4979 %       \@xdim := \@xdim + DELX * \unitlength
4980 %       \@ydim := \@ydim + DELY * \unitlength
4981 %     od
4982 %     \ignorespaces
4983 %   END
4984 %
4985 % \shortstack[POS]{TEXT} : Makes a \vbox containing TEXT stacked as
4986 %   a one-column array, positioned l, r or c as indicated by POS.
4987
4988 \newdimen\@wholewidth
4989 \newdimen\@halfwidth
4990 \newdimen\unitlength \unitlength =1pt
4991 \newbox\@picbox
4992 \newdimen\@picht
4993
4994 \def\picture(#1,#2){\ifnextchar({\@picture(#1,#2)}{\@picture(#1,#2)(0,0)}}
4995
4996 \def\@picture(#1,#2)(#3,#4){\@picht #2\unitlength
4997 \setbox\@picbox\hbox to#1\unitlength\bgroup
4998 \hskip -#3\unitlength \lower #4\unitlength \hbox\bgroup\ignorespaces}
4999
5000 %% 91/08/26 RmS & FMi: extra boxing level around \@picbox
5001 %%   to guard against unboxing in math mode
5002 %%   (proposed by John Hobby)
5003
5004 \def\endpicture{\egroup\hss\egroup\ht\@picbox\@picht
5005 \dp\@picbox\z@\mbox{\box\@picbox}}
5006
5007 % In the definitions of \put and \multiput, \hskip was replaced by \kern
5008 % just in case arg #3 = 'plus'. (Bug detected by Don Knuth.
5009 % changed 20 Jul 87).
5010 %
5011 \long\def\put(#1,#2)#3{\@killglue\raise#2\unitlength\hbox to\z@\kern
5012 #1\unitlength #3\hss}\ignorespaces}
5013
5014 \long\def\multiput(#1,#2)(#3,#4)#5#6{\@killglue\@multicnt #5\relax
5015 \@xdim #1\unitlength
5016 \@ydim #2\unitlength
5017 \@whilenum \@multicnt >\z@\do
5018 {\raise\@ydim\hbox to\z@\kern
5019 \@xdim #6\hss}\advance\@multicnt \m@ne\advance\@xdim
5020 #3\unitlength\advance\@ydim #4\unitlength}\ignorespaces}
5021
5022 \def\@killglue{\unskip\@whiledim \lastskip >\z@\do{\unskip}}
5023
5024 \def\thinlines{\let\@linefnt\tenln \let\@circlefnt\tencirc
5025 \@wholewidth\fontdimen8\tenln \@halfwidth .5\@wholewidth}
5026 \def\thicklines{\let\@linefnt\tenlnw \let\@circlefnt\tencircw
5027 \@wholewidth\fontdimen8\tenlnw \@halfwidth .5\@wholewidth}
5028
5029 \def\linethickness#1{\@wholewidth #1\relax \@halfwidth .5\@wholewidth}
5030

```

```

5031 \def\shortstack{\@ifnextchar[{\@shortstack}{\@shortstack[c]}}
5032
5033 \def\@shortstack[#1]{\leavevmode
5034 \vbox\bgroup\baselineskip-\p@\lineskip 3\p@\let\mb@1\hss
5035 \let\mb@r\hss \expandafter\let\csname mb@#1\endcsname\relax
5036 \let\\@stackcr\@ishortstack}
5037
5038 %% RmS 91/08/14 inserted extra braces around entry for NFSS
5039 \def\@ishortstack#1{\halign{\mb@1 {##}\unskip\mb@r\cr #1\cr}\egroup}
5040
5041
5042 \def\@stackcr{\@ifstar{\@ixstackcr}{\@ixstackcr}}
5043 \def\@ixstackcr{\@ifnextchar[{\@istackcr}{\cr\ignorespaces}}
5044
5045 \def\@istackcr[#1]{\cr\noalign{\vskip #1}\ignorespaces}
5046
5047
5048 % \line(X,Y){LEN} ==
5049 % BEGIN
5050 % \@xarg := X
5051 % \@yarg := Y
5052 % \@linelen := LEN * \unitlength
5053 % if \@xarg = 0
5054 % then \@uline
5055 % else if \@yarg = 0
5056 % then \@hline
5057 % else \@sline
5058 % if
5059 % if
5060 % END
5061 %
5062 % \@sline ==
5063 % BEGIN
5064 % if \@xarg < 0
5065 % then @negarg := T
5066 % \@xarg := -\@xarg
5067 % \@yyarg := -\@yarg
5068 % else @negarg := F
5069 % \@yyarg := \@yarg
5070 % fi
5071 % \@tempcnta := |\@yyarg|
5072 % if \@tempcnta > 6
5073 % then error: 'LATEX ERROR: Illegal \line or \vector argument.'
5074 % \@tempcnta := 0
5075 % fi
5076 % \box\@linechar := \hbox{\@linefnt \@getlinechar(\@xarg,\@yyarg) }
5077 % if \@yarg > 0 then \@upordown = \raise
5078 % \@clnht := 0
5079 % else \@upordown = \lower
5080 % \@clnht := height of \box\@linechar
5081 % fi
5082 % \@clnwd := width of \box\@linechar
5083 % if @negarg
5084 % then \hskip - width of \box\@linechar
5085 % \@tempa == \hskip - 2* width of box \@linechar
5086 % else \@tempa == \relax
5087 % fi
5088 %% Put out integral number of line segments
5089 % while \@clnwd < \@linelen
5090 % do \@upordown \@clnht \copy\@linechar
5091 % \@tempa
5092 % \@clnht := \@clnht + ht of \box\@linechar
5093 % \@clnwd := \@clnwd + width of \box\@linechar
5094 % od
5095 %
5096 %% Put out last segment
5097 % \@clnht := \@clnht - height of \box\@linechar
5098 % \@clnwd := \@clnwd - width of \box\@linechar
5099 % \@tempdima := \@linelen - \@clnwd
5100 % \@tempdimb := \@tempdima - width of \box\@linechar
5101 % if @negarg then \hskip -\@tempdimb
5102 % else \hskip \@tempdimb
5103 % fi
5104 % \@tempdima := 1000 * \@tempdima

```



```

5105 %   \@tempcnta := \@tempdima / width of \box\@linechar
5106 %   \@tempdima := (\@tempcnta * ht of \box\@linechar)/1000
5107 %   \@clnht := \@clnht + \@tempdima
5108 %   if \@linelen < width of \box\@linechar
5109 %       then \hskip width of \box\@linechar
5110 %       else \hbox{\@upordown \@clnht \copy\@linechar}
5111 %   fi
5112 % END
5113 %
5114 % \@hline ==
5115 % BEGIN
5116 %   if \@xarg < 0 then \hskip -\@linelen \fi
5117 %   \vrule height \@halfwidth depth \@halfwidth width \@linelen
5118 %   if \@xarg < 0 then \hskip -\@linelen \fi
5119 % END
5120 %
5121 % \@vline == if \@yarg < 0 \@downline else \@upline fi
5122 %
5123 %
5124 % \@getlinechar(X,Y) ==
5125 % BEGIN
5126 %   \@tempcnta := 8*X - 9
5127 %   if Y > 0
5128 %       then \@tempcnta := \@tempcnta + Y
5129 %       else \@tempcnta := \@tempcnta - Y + 64
5130 %   fi
5131 %   \char\@tempcnta
5132 % END
5133 %
5134 % \vector(X,Y){LEN} ==
5135 % BEGIN
5136 %   \@xarg := X
5137 %   \@yarg := Y
5138 %   \@linelen := LEN * \unitlength
5139 %   if \@xarg = 0
5140 %       then \@vvector
5141 %       else if \@yarg = 0
5142 %           then \@hvector
5143 %           else \@svector
5144 %       if
5145 %   if
5146 % END
5147 %
5148 % \@hvector ==
5149 % BEGIN
5150 %   \@hline
5151 %   {\@linefnt if \@xarg < 0 then \@getlarrow(1,0)
5152 %     else \@getrarrow(1,0)
5153 %   fi}
5154 % END
5155 %
5156 % \@vvector == if \@yarg < 0 \@downvector else \@upvector fi
5157 %
5158 % \@svector ==
5159 % BEGIN
5160 %   \@sline
5161 %   \@tempcnta := |\@yarg|
5162 %   if \@tempcnta < 5
5163 %       then \hskip - width of \box\@linechar
5164 %           \@upordown \@clnht \hbox
5165 %             {\@linefnt
5166 %               if @negarg then \@getlarrow(\@xarg,\@yyarg)
5167 %               else \@getrarrow(\@xarg,\@yyarg)
5168 %             fi }
5169 %       else error: 'LATEX ERROR: Illegal \line or \vector argument.'
5170 %   fi
5171 % END
5172 %
5173 % \@getlarrow(X,Y) ==
5174 % BEGIN
5175 %   if Y = 0
5176 %       then \@tempcnta := '33
5177 %       else \@tempcnta := 16 * X - 9
5178 %       \@tempcntb := 2 * Y

```

```

5179 %         if \@tempcntb > 0
5180 %           then \@tempcnta := \@tempcnta + \@tempcntb
5181 %           else \@tempcnta := \@tempcnta - \@tempcntb + 64
5182 %         fi
5183 %       fi
5184 %     \char\@tempcnta
5185 % END
5186 %
5187 % \@getrarrow(X,Y) ==
5188 % BEGIN
5189 %   \@tempcntb := |Y|
5190 %   case of \@tempcntb
5191 %     0 : \@tempcnta := '55
5192 %     1 : if X < 3
5193 %       then \@tempcnta := 24*X - 6
5194 %       else if X = 3
5195 %         then \@tempcnta := 49
5196 %         else \@tempcnta := 58 fi
5197 %     fi
5198 %     2 : if X < 3
5199 %       then \@tempcnta := 24*X - 3
5200 %       else \@tempcnta := 51 % X must = 3
5201 %     fi
5202 %     3 : \@tempcnta := 16*X - 2
5203 %     4 : \@tempcnta := 16*X + 7
5204 %   endcase
5205 %   if Y < 0
5206 %     then \@tempcnta := \@tempcnta + 64
5207 %   fi
5208 %   \char\@tempcnta
5209 % END
5210
5211 \newif\if@negarg
5212
5213 \def\line(#1,#2)#3{\@xarg #1\relax \@yarg #2\relax
5214 \@linelen #3\unitlength
5215 \ifnum\@xarg =\z@ \vline
5216   \else \ifnum\@yarg =\z@ \hline \else \sline\fi
5217 \fi}
5218
5219 \def\sline{\ifnum\@xarg<\z@ \@negargtrue \@xarg -\@xarg \@yyarg -\@yarg
5220   \else \@negargfalse \@yyarg \@yarg \fi
5221 \ifnum \@yyarg >\z@ \@tempcnta\@yyarg \else \@tempcnta -\@yyarg \fi
5222 \ifnum\@tempcnta>6 \@badlinearg\@tempcnta\z@ \fi
5223 \ifnum\@xarg>6 \@badlinearg\@xarg \@ne \fi
5224 \setbox\@linechar\hbox{\@linefont\@getlinechar(\@xarg,\@yyarg)}%
5225 \ifnum \@yarg >\z@ \let\@upordown\raise \@clnht\z@
5226   \else\let\@upordown\lower \@clnht \ht\@linechar\fi
5227 \@clnwd \wd\@linechar
5228 \if@negarg \hskip -\wd\@linechar \def\@tempa{\hskip -2\wd\@linechar}\else
5229   \let\@tempa\relax \fi
5230 \@whiledim \@clnwd <\@linelen \do
5231   {\@upordown\@clnht\copy\@linechar
5232   \@tempa
5233   \advance\@clnht \ht\@linechar
5234   \advance\@clnwd \wd\@linechar}%
5235 \advance\@clnht -\ht\@linechar
5236 \advance\@clnwd -\wd\@linechar
5237 \@tempdima\@linelen\advance\@tempdima -\@clnwd
5238 \@tempdimb\@tempdima\advance\@tempdimb -\wd\@linechar
5239 \if@negarg \hskip -\@tempdimb \else \hskip \@tempdimb \fi
5240 \multiply\@tempdima \@m
5241 \@tempcnta \@tempdima \@tempdima \wd\@linechar \divide\@tempcnta \@tempdima
5242 \@tempdima \ht\@linechar \multiply\@tempdima \@tempcnta
5243 \divide\@tempdima \@m
5244 \advance\@clnht \@tempdima
5245 \ifdim \@linelen <\wd\@linechar
5246   \hskip \wd\@linechar
5247   \else\@upordown\@clnht\copy\@linechar\fi}
5248
5249 \def\hline{\ifnum \@xarg <\z@ \hskip -\@linelen \fi
5250 \vrule \@height \@halfwidth \@depth \@halfwidth \@width \@linelen
5251 \ifnum \@xarg <\z@ \hskip -\@linelen \fi}
5252

```

```

5253 \def\@getlinechar(#1,#2){\@tempcnta#1\relax\multiply\@tempcnta 8%
5254 \advance\@tempcnta -9\ifnum #2>\z@ \advance\@tempcnta #2\relax\else
5255 \advance\@tempcnta -#2\relax\advance\@tempcnta 64 \fi
5256 \char\@tempcnta}
5257
5258 \def\vector(#1,#2)#3{\@xarg #1\relax \@yarg #2\relax
5259 \@tempcnta \ifnum\@xarg<\z@ -\@xarg\else\@xarg\fi
5260 \ifnum\@tempcnta<5\relax
5261 \@linelen #3\unitlength
5262 \ifnum\@xarg =\z@ \@vvector
5263 \else \ifnum\@yarg =\z@ \@hvector \else \@svector\fi
5264 \fi
5265 \else\@badlinearg\fi}
5266
5267 \def\@hvector{\@hline\hbox to\z@{\@linefnt
5268 \ifnum \@xarg <\z@ \@getlarrow(1,0)\hss\else
5269 \hss\@getrarrow(1,0)\fi}}
5270
5271 \def\@vvector{\ifnum \@yarg <\z@ \@downvector \else \@upvector \fi}
5272
5273 \def\@svector{\@sline
5274 \@tempcnta\@yarg \ifnum\@tempcnta <\z@ \@tempcnta -\@tempcnta\fi
5275 \ifnum\@tempcnta <5%
5276 \hskip -\wd\@linechar
5277 \@upordown\@clnht \hbox{\@linefnt \if@negarg
5278 \@getlarrow(\@xarg,\@yyarg)\else \@getrarrow(\@xarg,\@yyarg)\fi}%
5279 \else\@badlinearg\fi}
5280
5281 \def\@getlarrow(#1,#2){\ifnum #2=\z@ \@tempcnta'33 \else
5282 \@tempcnta #1\relax\multiply\@tempcnta \sixt@n \advance\@tempcnta
5283 -9 \@tempcntb #2\relax\multiply\@tempcntb \tw@
5284 \ifnum \@tempcntb >\z@ \advance\@tempcnta \@tempcntb
5285 \else\advance\@tempcnta -\@tempcntb\advance\@tempcnta 64
5286 \fi\fi\char\@tempcnta}
5287
5288 \def\@getrarrow(#1,#2){\@tempcntb #2\relax
5289 \ifnum\@tempcntb <\z@ \@tempcntb -\@tempcntb\relax\fi
5290 \ifcase \@tempcntb\relax \@tempcnta'55 \or
5291 \ifnum #1<\thr@@ \@tempcnta #1\relax\multiply\@tempcnta
5292 24\advance\@tempcnta -6 \else \ifnum #1=\thr@@ \@tempcnta 49
5293 \else\@tempcnta 58 \fi\fi\or
5294 \ifnum #1<\thr@@ \@tempcnta=#1\relax\multiply\@tempcnta
5295 24\advance\@tempcnta -\thr@@ \else \@tempcnta 51 \fi\or
5296 \@tempcnta #1\relax\multiply\@tempcnta
5297 \sixt@n \advance\@tempcnta -\tw@ \else
5298 \@tempcnta #1\relax\multiply\@tempcnta
5299 \sixt@n \advance\@tempcnta 7 \fi\ifnum #2<\z@ \advance\@tempcnta 64 \fi
5300 \char\@tempcnta}
5301
5302
5303
5304 \def\@vline{\ifnum \@yarg <\z@ \@downline \else \@upline\fi}
5305
5306 \def\@upline{\hbox to \z@{\hskip -\@halfwidth \vrule \@width \@wholewidth
5307 \@height \@linelen \@depth \z@\hss}}
5308
5309 \def\@downline{\hbox to \z@{\hskip -\@halfwidth \vrule \@width \@wholewidth
5310 \@height \z@ \@depth \@linelen \hss}}
5311
5312 \def\@upvector{\@upline\setbox\@tempboxa\hbox{\@linefnt\char'66}\raise
5313 \@linelen \hbox to\z@{\lower \ht\@tempboxa\box\@tempboxa\hss}}
5314
5315 \def\@downvector{\@downline\lower \@linelen
5316 \hbox to \z@{\@linefnt\char'77\hss}}
5317
5318 % \dashbox{D}(X,Y) ==
5319 % BEGIN
5320 % leave vertical mode
5321 % \hbox to Opt {
5322 % \baselineskip := Opt
5323 % \lineskip := Opt
5324 % %% HORIZONTAL DASHES
5325 % \@dashdim := X * \unitlength
5326 % \@dashcnt := \@dashdim + 200 % to prevent roundoff error

```

```

5327 % \dashdim := D * \unitlength
5328 % \dashcnt := \dashcnt / \dashdim
5329 % if \dashcnt is odd
5330 % then \dashdim := Opt
5331 % \dashcnt := (\dashcnt + 1) / 2
5332 % else \dashdim := \dashdim / 2
5333 % \dashcnt := \dashcnt / 2 - 1
5334 % \box\@dashbox := \hbox{\vrule height \@halfwidth
5335 % depth \@halfwidth width \dashdim}
5336 % \put(0,0){\copy\@dashbox}
5337 % \put(0,Y){\copy\@dashbox}
5338 % \put(X,0){\hskip -\dashdim\copy\@dashbox}
5339 % \put(X,Y){\hskip -\dashdim\box\@dashbox}
5340 % \dashdim := 3 * \dashdim
5341 % fi
5342 % \box\@dashbox := \hbox{\vrule height \@halfwidth
5343 % depth \@halfwidth width D * \unitlength
5344 % \hskip D * \unitlength}
5345 % \@tempcnta := 0
5346 % \put(0,0){\hskip \dashdim
5347 % while \@tempcnta < \dashcnt
5348 % do \copy\@dashbox
5349 % \@tempcnta := \@tempcnta + 1
5350 % od
5351 % }
5352 % \@tempcnta := 0
5353 % \put(0,Y){\hskip \dashdim
5354 % while \@tempcnta < \dashcnt
5355 % do \copy\@dashbox
5356 % \@tempcnta := \@tempcnta + 1
5357 % od
5358 % }
5359 %
5360 %% vertical dashes
5361 % \dashdim := Y * \unitlength
5362 % \dashcnt := \dashdim + 200 % to prevent roundoff error
5363 % \dashdim := D * \unitlength
5364 % \dashcnt := \dashcnt / \dashdim
5365 % if \dashcnt is odd
5366 % then \dashdim := Opt
5367 % \dashcnt := (\dashcnt + 1) / 2
5368 % else \dashdim := \dashdim / 2
5369 % \dashcnt := \dashcnt / 2 - 1
5370 % \box\@dashbox := \hbox{\hskip -\@halfwidth
5371 % \vrule width \@wholewidth
5372 % height \dashdim }
5373 % \put(0,0){\copy\@dashbox}
5374 % \put(X,0){\copy\@dashbox}
5375 % \put(0,Y){\lower\@dashdim\copy\@dashbox}
5376 % \put(X,Y){\lower\@dashdim\copy\@dashbox}
5377 % \dashdim := 3 * \dashdim
5378 % fi
5379 % \box\@dashbox := \hbox{\vrule width \@wholewidth
5380 % height D * \unitlength }
5381 % \@tempcnta := 0
5382 % \put(0,0){\hskip -\@halfwidth
5383 % \vbox{while \@tempcnta < \dashcnt
5384 % do \vskip D*\unitlength
5385 % \copy\@dashbox
5386 % \@tempcnta := \@tempcnta + 1
5387 % od
5388 % \vskip \dashdim
5389 % } }
5390 % \@tempcnta := 0
5391 % \put(X,0){\hskip -\@halfwidth
5392 % \vbox{while \@tempcnta < \dashcnt
5393 % do \vskip D*\unitlength
5394 % \copy\@dashbox
5395 % \@tempcnta := \@tempcnta + 1
5396 % od
5397 % \vskip \dashdim
5398 % } }
5399 % }
5400 % } % END DASHES

```

```

5401 %
5402 % \@makepicbox(X,Y)
5403 % END
5404
5405 \def\dashbox#1(#2,#3){\leavevmode\hbox to\z@{\baselineskip \z@
5406 \lineskip \z@
5407 \@dashdim #2\unitlength
5408 \@dashcnt \@dashdim \advance\@dashcnt 200
5409 \@dashdim #1\unitlength\divide\@dashcnt \@dashdim
5410 \ifodd\@dashcnt\@dashdim \z@
5411 \advance\@dashcnt \one \divide\@dashcnt \tw@
5412 \else \divide\@dashdim \tw@ \divide\@dashcnt \tw@
5413 \advance\@dashcnt \m@ne
5414 \setbox\@dashbox \hbox{\vrule \@height \@halfwidth \@depth \@halfwidth
5415 \@width \@dashdim}\put(0,0){\copy\@dashbox}%
5416 \put(0,#3){\copy\@dashbox}%
5417 \put(#2,0){\hskip-\@dashdim\copy\@dashbox}%
5418 \put(#2,#3){\hskip-\@dashdim\box\@dashbox}%
5419 \multiply\@dashdim \thr@@
5420 \fi
5421 \setbox\@dashbox \hbox{\vrule \@height \@halfwidth \@depth \@halfwidth
5422 \@width #1\unitlength\hskip #1\unitlength}\@tempcnta\z@
5423 \put(0,0){\hskip\@dashdim \@whilenum \@tempcnta <\@dashcnt
5424 \do{\copy\@dashbox\advance\@tempcnta \one }}\@tempcnta\z@
5425 \put(0,#3){\hskip\@dashdim \@whilenum \@tempcnta <\@dashcnt
5426 \do{\copy\@dashbox\advance\@tempcnta \one }}%
5427 \@dashdim #3\unitlength
5428 \@dashcnt \@dashdim \advance\@dashcnt 200
5429 \@dashdim #1\unitlength\divide\@dashcnt \@dashdim
5430 \ifodd\@dashcnt \@dashdim \z@
5431 \advance\@dashcnt \one \divide\@dashcnt \tw@
5432 \else
5433 \divide\@dashdim \tw@ \divide\@dashcnt \tw@
5434 \advance\@dashcnt \m@ne
5435 \setbox\@dashbox\hbox{\hskip -\@halfwidth
5436 \vrule \@width \@wholewidth
5437 \@height \@dashdim}\put(0,0){\copy\@dashbox}%
5438 \put(#2,0){\copy\@dashbox}%
5439 \put(0,#3){\lower\@dashdim\copy\@dashbox}%
5440 \put(#2,#3){\lower\@dashdim\copy\@dashbox}%
5441 \multiply\@dashdim \thr@@
5442 \fi
5443 \setbox\@dashbox\hbox{\vrule \@width \@wholewidth
5444 \@height #1\unitlength}\@tempcnta\z@
5445 \put(0,0){\hskip -\@halfwidth \vbox{\@whilenum \@tempcnta <\@dashcnt
5446 \do{\vskip #1\unitlength\copy\@dashbox\advance\@tempcnta \one }}%
5447 \vskip\@dashdim}}\@tempcnta\z@
5448 \put(#2,0){\hskip -\@halfwidth \vbox{\@whilenum \@tempcnta <\@dashcnt
5449 \do{\vskip #1\unitlength\copy\@dashbox\advance\@tempcnta \one }}%
5450 \vskip\@dashdim}}\@makepicbox(#2,#3)}
5451
5452 % CIRCLES AND OVALS
5453 %
5454 % USER COMMANDS:
5455 %
5456 % \circle{D} : Produces the circle with the diameter as close as
5457 %             possible to D * \unitlength. \put(X,Y){\circle{D}}
5458 %             puts the circle with its center at (X,Y).
5459 %
5460 % \oval(X,Y) : Makes an oval as round as possible that fits in the
5461 %             rectangle of width X * \unitlength and height
5462 %             Y * \unitlength. The reference point is the center.
5463 %
5464 % \oval(X,Y)[POS] : Save as \oval(X,Y) except it draws only the
5465 %                 half or quadrant of the oval indicated by POS.
5466 %                 E.G., \oval(X,Y)[t] draws just the top half
5467 %                 and \oval(X,Y)[br] draws just the bottom right
5468 %                 quadrant. In all cases, the reference point is
5469 %                 the same as the unqualified \oval(X,Y) command.
5470 %
5471 % \@overt {DELTA1} {DELTA2} : Makes a vbox containing either the left side
5472 %                             or the right side of the oval being constructed. The baseline
5473 %                             will coincide with the outside bottom edge of the oval; the left
5474 %                             side of the box will coincide with the left edge of the vertical

```

```

5475 % rule. The width of the box will be \@tempdima.
5476 % DELTA1 and DELTA2 are added to the character number in \@tempcnta
5477 % to get the characters for the top and bottom quarter circle pieces.
5478 %
5479 % \@ovhorz : Makes an hbox containing the straight rule for either the
5480 % top or the bottom of the oval being constructed. The baseline
5481 % will coincide with bottom edge of the rule; the left side of
5482 % the box will coincide with the left side of the oval.
5483 % The width of the box will be \@ovxx.
5484 %
5485 % \@getcirc {DIAM} : Sets \@tempcnta to the character number
5486 % of the top-right quarter circle with the largest
5487 % diameter less than or equal to DIAM.
5488 % Sets \@tempboxa to an hbox containing that character.
5489 % Sets \@tempdima to \wd \@tempboxa, which is the distance
5490 % from the circle's left outside edge to its right
5491 % inside edge.
5492 % (These characters are like those described in the
5493 % TeXbook, pp. 389-90.)
5494 %
5495 % \@getcirc {DIAM} ==
5496 % BEGIN
5497 % \@tempcnta := integer coercion of (DIAM + 2pt) %% + 2pt added
5498 % \@tempcnta := \@tempcnta / integer coercion of 4pt %% 1 Nov 88
5499 % if \@tempcnta > 10
5500 % then \@tempcnta := 10 fi
5501 % if \@tempcnta > 0
5502 % then \@tempcnta := \@tempcnta-1
5503 % else LaTeX Warning: Oval too small.
5504 % fi
5505 % \@tempcnta := 4 * \@tempcnta
5506 % \@tempboxa := \hbox{\@circlefnt \char \@tempcnta}
5507 % \@tempdima := \wd \@tempboxa
5508 % END
5509 %
5510 % \@put{X}{Y}{OBJ} ==
5511 % BEGIN
5512 % \raise Y \hbox to Opt{\hskip X OBJ \hss}
5513 % END
5514 %
5515 % \@oval(X,Y)[POS] ==
5516 % BEGIN
5517 % \begingroup
5518 % \boxmaxdepth := \maxdimen
5519 % @out := @ovb := @ovl := @ovr := true
5520 % for all E in POS
5521 % do @ovE := false od
5522 % \@ovxx := X * \unitlength
5523 % \@ovyy := Y * \unitlength
5524 % \@tempdimb := min(\@ovxx, \@ovyy)
5525 % \@getcirc{\@tempdimb-2pt} %% "-2pt" added 7 Dec 89
5526 % \@ovro := \ht \@tempboxa
5527 % \@ovri := \dp \@tempboxa
5528 % \@ovdx := \@ovxx - \@tempdima
5529 % \@ovdy := \@ovdx/2
5530 % \@ovdy := \@ovyy - \@tempdima
5531 % \@ovdy := \@ovdy/2
5532 % \@circlefnt
5533 % \@tempboxa :=
5534 % \hbox{
5535 % if @ovr
5536 % then \@overt{3}{2} \kern -\@tempdima
5537 % fi
5538 % if @ovl
5539 % then \kern \@ovxx \@overt{0}{1} \kern -\@tempdima
5540 % \kern -\@ovxx
5541 % fi
5542 % if @out
5543 % then \@ovhorz \kern -\@ovxx
5544 % fi
5545 % if @ovb
5546 % then \raise \@ovyy \@ovhorz
5547 % fi
5548 % }

```

```

5549 %      \@ovdx := \@ovdx + \@ouro
5550 %      \@ovdy := \@ovdy + \@ouro
5551 %      \ht\@tempboxa := \dp\@tempboxa := 0
5552 %      \put{-\@ovdx}{-\@ovdy}{\box\@tempboxa}
5553 %    \endgroup
5554 %  END
5555 %
5556 % \overt {DELTA1} {DELTA2} ==
5557 % BEGIN
5558 %   \vbox to \ovvy {
5559 %     if @ovb
5560 %       then \@tempcntb := \@tempcnta + DELTA1
5561 %         \kern -\@ouro
5562 %         \hbox { \char \@tempcntb }
5563 %         \nointerlineskip
5564 %       else \kern \@ovri \kern \@ovdy
5565 %     fi
5566 %     \leaders \vrule width \@wholewidth \vfil
5567 %     \nointerlineskip
5568 %     if @out
5569 %       then \@tempcntb := \@tempcnta + DELTA2
5570 %         \hbox { \char \@tempcntb }
5571 %       else \kern \@ovdy \kern \@ouro
5572 %     fi
5573 %   }
5574 % END
5575 %
5576 % \ovhorz ==
5577 % BEGIN
5578 %   \hbox to \@ovx{
5579 %     \kern \@ouro
5580 %     if @ovr
5581 %       then
5582 %         else \kern \@ovdx
5583 %       fi
5584 %     \leaders \hrule height \@wholewidth \hfil
5585 %     if @ovl
5586 %       then
5587 %         else \kern \@ovdx
5588 %       fi
5589 %     \kern \@ovri
5590 %   }
5591 % END
5592 %
5593 % \circle{DIAM} ==
5594 % BEGIN
5595 %   \begingroup
5596 %   \boxmaxdepth := maxdimen
5597 %   \@tempdimb := DIAM * \unitlength
5598 %   if \@tempdimb > 15.5pt
5599 %     then \getcirc{\@tempdimb}
5600 %         \@ouro := \ht \@tempboxa
5601 %         \@tempboxa := \hbox{
5602 %           \@circfnt
5603 %           \@tempcnta := \@tempcnta + 2
5604 %           \char \@tempcnta
5605 %           \@tempcnta := \@tempcnta - 1
5606 %           \char \@tempcnta
5607 %           \kern -2\@tempdima
5608 %           \@tempcnta := \@tempcnta + 2
5609 %           \raise \@tempdima \hbox { \char \@tempcnta }
5610 %           \raise \@tempdima \box\@tempboxa
5611 %         }
5612 %     \ht\@tempboxa := \dp\@tempboxa := 0
5613 %     \put{-\@ouro}{-\@ouro}{\@tempboxa}
5614 %   else
5615 %     \@circ{\@tempdimb}{96}
5616 %   fi
5617 % \endgroup
5618 % END
5619 %
5620 % \circle*{DIAM} == \dot{DIAM} == \circ{DIAM*\unitlength}{112}
5621 %
5622 % \circ{DIAM}{CHAR} ==

```

```

5623 % BEGIN
5624 % \@tempcnta := integer coercion of (DIAM + .5pt)/1pt.
5625 % if \@tempcnta > 15 then \@tempcnta := 15 fi
5626 % if \@tempcnta > 1 then \@tempcnta := \@tempcnta - 1 fi
5627 % \@tempcnta := \@tempcnta + CHAR
5628 % \@circlefnt
5629 % \char \@tempcnta
5630 % END
5631 %
5632
5633 \newif\if@ovt
5634 \newif\if@ovb
5635 \newif\if@ovl
5636 \newif\if@ovr
5637 \newdimen\@ovxx
5638 \newdimen\@ovyy
5639 \newdimen\@ovdx
5640 \newdimen\@ovdy
5641 \newdimen\@ovro
5642 \newdimen\@ovri
5643
5644 %% \advance\@tempdima 2pt\relax added 1 Nov 88 to fix bug in which
5645 %% size of drawn circle not monotonic function of argument of \circle,
5646 %% caused by different rounding for dimensions of large and small circles.
5647 %
5648 \def\@getcirc#1{\@tempdima #1\relax \advance\@tempdima 2\p@
5649 \@tempcnta\@tempdima
5650 \@tempdima 4\p@ \divide\@tempcnta\@tempdima
5651 \ifnum \@tempcnta >10\relax \@tempcnta 10\relax\fi
5652 \ifnum \@tempcnta >\z@ \advance\@tempcnta\m@ne
5653 \else \warning{Oval too small}\fi
5654 \multiply\@tempcnta 4\relax
5655 \setbox \@tempboxa \hbox{\@circlefnt
5656 \char \@tempcnta}\@tempdima \wd \@tempboxa}
5657
5658 \def\@put#1#2#3{\raise #2\hbox to\z@{\hskip #1#3\hss}}
5659
5660 \def\@oval(#1,#2){\ifnextchar[{\@oval(#1,#2)}{\@oval(#1,#2) []}}
5661
5662 \def\@oval(#1,#2)[#3]{\begingroup\boxmaxdepth \maxdimen
5663 \@ovttrue \@ovbtrue \@ovltrue \@ovrtrue
5664 \@tfor\@tempa :=#3\do{\csname @ov\@tempa false\endcsname}\@ovxx
5665 #1\unitlength \@ovyy #2\unitlength
5666 \@tempdimb \ifidim \@ovyy >\@ovxx \@ovxx\else \@ovyy \fi
5667 \advance \@tempdimb -2\p@
5668 \@getcirc \@tempdimb
5669 \@ovro \ht\@tempboxa \@ovri \dp\@tempboxa
5670 \@ovdx\@ovxx \advance\@ovdx -\@tempdima \divide\@ovdx \tw@
5671 \@ovdy\@ovyy \advance\@ovdy -\@tempdima \divide\@ovdy \tw@
5672 \@circlefnt \setbox\@tempboxa
5673 \hbox{\if@ovr \@ovvert32\kern -\@tempdima \fi
5674 \if@ovl \kern \@ovxx \@ovvert01\kern -\@tempdima \kern -\@ovxx \fi
5675 \if@ovt \@ovhorz \kern -\@ovxx \fi
5676 \if@ovb \raise \@ovyy \@ovhorz \fi}\advance\@ovdx\@ovro
5677 \advance\@ovdy\@ovro \ht\@tempboxa\z@ \dp\@tempboxa\z@
5678 \@put{-\@ovdx}{-\@ovdy}{\box\@tempboxa}%
5679 \endgroup}
5680
5681 \def\@ovvert#1#2{\vbox to\@ovyy{%
5682 \if@ovb \@tempcntb \@tempcnta \advance \@tempcntb #1\relax
5683 \kern -\@ovro \hbox{\char \@tempcntb}\nointerlineskip
5684 \else \kern \@ovri \kern \@ovdy \fi
5685 \leaders\vrule \@width \@wholewidth\vfil \nointerlineskip
5686 \if@ovt \@tempcntb \@tempcnta \advance \@tempcntb #2\relax
5687 \hbox{\char \@tempcntb}%
5688 \else \kern \@ovdy \kern \@ovro \fi}}
5689
5690 \def\@ovhorz{\hbox to \@ovxx{\kern \@ovro
5691 \if@ovr \else \kern \@ovdx \fi
5692 \leaders\hrule \@height \@wholewidth \hfil
5693 \if@ovl \else \kern \@ovdx \fi
5694 \kern \@ovri}}
5695
5696 \def\circle{\@ifstar{\@dot}{\@circle}}

```



```

5697 \def\@circ#1{\begingroup \boxmaxdepth \maxdimen \@tempdimb #1\unitlength
5698   \ifdim \@tempdimb >15.5\p@ \getcirc\@tempdimb
5699   \ovro\ht\@tempboxa
5700   \setbox\@tempboxa\hbox{\@circleft
5701     \advance\@tempcnta\tw@ \char \@tempcnta
5702     \advance\@tempcnta\m@ne \char \@tempcnta \kern -2\@tempdima
5703     \advance\@tempcnta\tw@
5704     \raise \@tempdima \hbox{\char\@tempcnta}\raise \@tempdima
5705     \box\@tempboxa}\ht\@tempboxa\z@ \dp\@tempboxa\z@
5706     \put{-\@ovro}{-\@ovro}{\box\@tempboxa}%
5707   \else \@circ\@tempdimb{96}\fi\endgroup}
5708
5709 \def\@dot#1{\@tempdimb #1\unitlength \@circ\@tempdimb{112}}
5710
5711 \def\@circ#1#2{\@tempdima #1\relax \advance\@tempdima .5\p@
5712   \@tempcnta\@tempdima \@tempdima \p@
5713   \divide\@tempcnta\@tempdima
5714   \ifnum\@tempcnta >15\relax \@tempcnta 15\relax \fi
5715   \ifnum \@tempcnta >\z@ \advance\@tempcnta\m@ne\fi
5716   \advance\@tempcnta #2\relax
5717   \@circleft \char\@tempcnta}
5718
5719
5720 %INITIALIZATION
5721 \thinlines
5722
5723 \newcount\@xarg
5724 \newcount\@yarg
5725 \newcount\@yyarg
5726 \newcount\@multicnt
5727 \newdimen\@xdim
5728 \newdimen\@ydim
5729 \newbox\@linechar
5730 \newdimen\@linelen
5731 \newdimen\@clnwd
5732 \newdimen\@clnht
5733 \newdimen\@dashdim
5734 \newbox\@dashbox
5735 \newcount\@dashcnt
5736
5737
5738
5739 \message{theorem,}
5740 % *****
5741 % * THEOREM ENVIRONMENTS *
5742 % *****
5743 %
5744 % The user creates his own theorem-like environments with the command
5745 % \newtheorem{NAME}{TEXT}[COUNTER] or
5746 % \newtheorem{NAME}[OLDNAME]{TEXT}
5747 % This defines the environment NAME to be just as one would expect a
5748 % theorem environment to be, except that it prints ‘TEXT’ instead of
5749 % ‘Theorem’.
5750 %
5751 % If OLDNAME is given, then environments NAME and OLDNAME use the same
5752 % counter, so using a NAME environment advances the number of the next
5753 % NAME environment, and vice-versa.
5754 %
5755 % If COUNTER is given, then environment NAME is numbered within COUNTER.
5756 % E.g., if COUNTER = subsection, then the first NAME in subsection 7.2
5757 % is numbered TEXT 7.2.1.
5758 %
5759 % The way NAME environments are numbered can be changed by redefining
5760 % \theNAME.
5761 %
5762 % DOCUMENT STYLE PARAMETERS
5763 %
5764 % \@thmcounter{COUNTER} : A command such that
5765 % \edef\theCOUNTER{\@thmcounter{COUNTER}}
5766 % defines \theCOUNTER to produce a number for a theorem environment.
5767 % The default is:
5768 % BEGIN \noexpand\arabic{COUNTER} END
5769 %
5770 % \@thmcountersep : A separator placed between a theorem number and

```

```

5771 %      the number of the counter within which it is numbered.
5772 %      E.g., to make the third theorem of section 7.2 be numbered
5773 %      7.2-3, \thmcountersep should be \def'ed to '-'. Its
5774 %      default is '.'.
5775 %
5776 % \begintheorem{NAME}{NUMBER} : A command that begins a theorem
5777 %      environment for a 'theorem' named 'NAME NUMBER' --
5778 %      e.g., \begintheorem{Lemma}{3.7} starts Lemma 3.7.
5779 %
5780 % \opargbegintheorem{NAME}{NUMBER}{OPARG} : A command that begins a theorem
5781 %      environment for a 'theorem' named 'NAME NUMBER' with optional
5782 %      argument OPARG -- e.g., \opargbegintheorem{Lemma}{3.7}{Jones}
5783 %      starts 'Lemma 3.7 (Jones)'.
5784 %
5785 % \endtheorem : A command that ends a theorem environment.
5786 %
5787 % \newtheorem{NAME}{TEXT}[COUNTER] ==
5788 % BEGIN
5789 %   if \NAME is definable
5790 %     then \definecounter{NAME}
5791 %       if COUNTER present
5792 %         then \addtoreset{NAME}{COUNTER} fi
5793 %         \theNAME == BEGIN \theCOUNTER \thmcountersep
5794 %           eval\thmcounter{NAME} END
5795 %         else \theNAME == BEGIN eval\thmcounter{NAME} END
5796 %         \NAME == \thm{NAME}{TEXT}
5797 %         \endNAME == \endtheorem
5798 %     else error
5799 %   fi
5800 % END
5801 %
5802 % \newtheorem{NAME}[OLDNAME]{TEXT}==
5803 % BEGIN
5804 %   if \NAME is definable
5805 %     then \theNAME == \theOLDNAME
5806 %         \NAME == \thm{OLDNAME}{TEXT}
5807 %         \endNAME == \endtheorem
5808 %     else error
5809 %   fi
5810 % END
5811 %
5812 % \thm{NAME}{TEXT} ==
5813 % BEGIN
5814 %   \refstepcounter{NAME}
5815 %   if next char = [
5816 %     then \@ythm{NAME}{TEXT}
5817 %     else \@xthm{NAME}{TEXT}
5818 %   fi
5819 % END
5820 %
5821 % \@xthm{NAME}{TEXT} ==
5822 % BEGIN
5823 %   \begintheorem{TEXT}{\theNAME}
5824 %   \ignorespaces
5825 % END
5826 %
5827 % \@ythm{NAME}{TEXT}[OPARG] ==
5828 % BEGIN
5829 %   \opargbegintheorem{TEXT}{\theNAME}{OPARG}
5830 %   \ignorespaces
5831 % END
5832 %
5833 % \def\newtheorem#1{\ifnextchar[{\@othm{#1}}{\@nthm{#1}}}
5834 %
5835 % \def\@nthm#1#2{%
5836 % \ifnextchar[{\@xnthm{#1}{#2}}{\@ynthm{#1}{#2}}}
5837 %
5838 % \def\@xnthm#1#2[#3]{\expandafter\@ifdefinable\csname #1\endcsname
5839 % {\@definecounter{#1}\@addtoreset{#1}{#3}}%
5840 % \expandafter\xdef\csname the#1\endcsname{\expandafter\noexpand
5841 % \csname the#3\endcsname \thmcountersep \thmcounter{#1}}%
5842 % \global\@namedef{#1}{\@thm{#1}{#2}}\global\@namedef{end#1}{\endtheorem}}
5843 %
5844 % \def\@ynthm#1#2{\expandafter\@ifdefinable\csname #1\endcsname

```

```

5845 {\@definecounter{#1}%
5846 \expandafter\xdef\cename the#1\endcsname{\@thmcounter{#1}}%
5847 \global\@namedef{#1}{\@thm{#1}{#2}}\global\@namedef{end#1}{\@endtheorem}}
5848
5849 %% RmS 92/01/10: check for existence of theorem environment #2
5850 \def\@othm#1[#2]#3{%
5851   \@ifundefined{c#2}{\@latexerr{No theorem environment ‘#2’ defined}\@eha}%
5852   {\expandafter\@ifdefinable\cename #1\endcsname
5853     {\global\@namedef{the#1}{\@nameuse{the#2}}}%
5854   \global\@namedef{#1}{\@thm{#2}{#3}}%
5855   \global\@namedef{end#1}{\@endtheorem}}}}
5856
5857 \def\@thm#1#2{\refstepcounter
5858   {#1}\@ifnextchar[{\@ythm{#1}{#2}}{\@xthm{#1}{#2}}}
5859
5860 \def\@xthm#1#2{\@begintheorem{#2}{\cename the#1\endcsname}\ignorespaces}
5861 \def\@ythm#1#2[#3]{\@opargbegintheorem{#2}{\cename
5862   the#1\endcsname}{#3}\ignorespaces}
5863
5864 %DEFAULT VALUES
5865 \def\@thmcounter#1{\noexpand\arabic{#1}}
5866 \def\@thmcountersep{.}
5867 %deleted September 2, 1986 MDK
5868 %\def\@makethmnumber#1#2{\bf #1 #2:}
5869
5870 %% RmS 91/08/14 Moved \it after \item to make it work with NFSS
5871 \def\@begintheorem#1#2{\trivlist \item[\hskip \labelsep{\bf #1 #2}]\it}
5872 \def\@opargbegintheorem#1#2#3{\trivlist
5873   \item[\hskip \labelsep{\bf #1 #2\ (#3)}]\it}
5874 \def\@endtheorem{\endtrivlist}
5875
5876
5877 \message{lengths,}
5878 % *****
5879 % *          LENGTHS          *
5880 % *****
5881 %
5882 % USER COMMANDS:
5883 %
5884 % \newlength{NAME}      == \newskip NAME
5885 % \setlength{NAME}{VALUE} == NAME :=L VALUE
5886 % \addtolength{NAME}{VALUE} == NAME :=L NAME + VALUE
5887 % \settowidth{NAME}{TEXT} == NAME :=L width of \hbox{TEXT}
5888 %
5889 \def\newlength#1{\@ifdefinable#1{\newskip#1}}
5890 \def\setlength#1#2{\relax}
5891 \def\addtolength#1#2{\advance#1 #2\relax}
5892 \def\settowidth#1#2{\setbox\@tempboxa\hbox{#2}\wd\@tempboxa\relax}
5893 %% \relax added 24 Mar 86
5894
5895 \message{title,}
5896 % *****
5897 % *          THE TITLE          *
5898 % *****
5899 %
5900 % The user defines the title, author, date by the declarations \title{NAME},
5901 % \author{NAME} and \date{DATE}. Inside these, he can use the \thanks
5902 % command to make footnoted acknowledgements, notice of address, etc.
5903 % The \maketitle command produces the actual title. Note: multiple authors
5904 % are separated with the \and command.
5905
5906 \def\title#1{\gdef\@title{#1}}
5907
5908 \def\author#1{\gdef\@author{#1}}
5909
5910 \def\date#1{\gdef\@date{#1}}
5911 \gdef\@date{\today} %Default is today's date
5912
5913 \def\thanks#1{\footnotemark\beginngroup
5914 \def\protect{\noexpand\protect\noexpand}\xdef\@thanks{\@thanks
5915   \protect\footnotetext[\the\c@footnote]{#1}}\endngroup}
5916
5917 \def\@thanks{}
5918

```

```

5919 \def\andf%%                                % \begin{tabular}
5920 \end{tabular}\hskip 1em plus.17fil\begin{tabular}[t]{c}%% \end{tabular}
5921 }
5922
5923
5924
5925 \message{sectioning,}
5926 % *****
5927 % * SECTIONING *
5928 % *****
5929 %
5930 %
5931 % \@startsection {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}
5932 % optional * [ALTHEADING]{HEADING}
5933 % Generic command to start a section.
5934 % NAME : e.g., 'subsection'
5935 % LEVEL : a number, denoting depth of section -- e.g., chapter=1,
5936 % section = 2, etc.
5937 % INDENT : Indentation of heading from left margin
5938 % BEFORESKIP : Absolute value = skip to leave above the heading.
5939 % If negative, then paragraph indent of text following
5940 % heading is suppressed.
5941 % AFTERSKIP : if positive, then skip to leave below heading, else
5942 % negative of skip to leave to right of run-in heading.
5943 % STYLE : commands to set style
5944 % If '*' missing, then increments the counter. If it is present, then
5945 % there should be no [ALTHEADING] argument.
5946 % Uses the counter 'secnumdepth' whose value is the highest section
5947 % level that is to be numbered.
5948 %
5949 % WARNING: The \@startsection command should be at the same or higher
5950 % grouping level as the text that follows it. For example, you
5951 % should NOT do something like
5952 % \def\foo{ \begingroup ...
5953 % \paragraph{...}
5954 % \endgroup}
5955 %
5956 % \@startsection {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE} ==
5957 % BEGIN
5958 % IF @noskipsec = T THEN \leavevmode FI % true if previous section
5959 % % had no body.
5960 %
5961 % \par
5962 % \@tempskipa := BEFORESKIP
5963 % @afterindent := T
5964 % IF \@tempskipa < 0 THEN \@tempskipa := -\@tempskipa
5965 % @afterindent := F
5966 %
5967 % FI
5968 % IF @nobreak = true
5969 % THEN \everypar == null
5970 % ELSE \addpenalty{\@secpenalty}
5971 % \addvspace{\@tempskipa}
5972 %
5973 % FI
5974 % IF * next
5975 % THEN \ssect{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}
5976 % ELSE \dblarg{\sect
5977 % {NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}}
5978 %
5979 % FI
5980 % END
5981 %
5982 % \@sect{NAME}{LEVEL}{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}[ARG1]{ARG2} ==
5983 % BEGIN
5984 % IF LEVEL > \c@secnumdepth
5985 % THEN \@svsec :=L null
5986 % ELSE \refstepcounter{NAME}
5987 % \@svsec :=L BEGIN \theNAME END
5988 %
5989 % FI
5990 % IF AFTERSKIP > 0
5991 % THEN \begingroup
5992 % STYLE
5993 % \@hangfrom{\hskip INDENT\@svsec}
5994 % {\interlinepenalty 1000 ARG2\par}
5995 % \endgroup
5996 % \NAMEmark{ARG1}
5997 % \addcontentsline{toc}{NAME}

```

```

5993 %           { IF LEVEL > \c@secnumdepth
5994 %             ELSE \protect\numberline{\theNAME} FI
5995 %           ARG1 }
5996 % ELSE \@susechd == BEGIN STYLE
5997 %           \hskip INDENT\@susec
5998 %           ARG2
5999 %           \NAMEmark{ARG1}
6000 %           \addcontentsline{toc}{NAME}
6001 %           { IF LEVEL > \c@secnumdepth
6002 %             ELSE \protect\numberline{\theNAME} FI
6003 %           ARG1 }
6004 %           END
6005 % FI
6006 % \@xsect{AFTERSKIP}
6007 % END
6008 %
6009 % \@xsect{AFTERSKIP} ==
6010 % BEGIN
6011 % IF AFTERSKIP > 0
6012 % THEN \par \nobreak
6013 %       \vskip AFTERSKIP
6014 %       \@afterheading
6015 % ELSE @nobreak :=G F
6016 %       @noskipsec :=G T
6017 %       \everypar{ IF @noskipsec = T
6018 %                 THEN @noskipsec :=G F
6019 %                 \clubpenalty :=G 10000
6020 %                 \hskip -\parindent
6021 %                 \begingroup
6022 %                 \@susechd
6023 %                 \endgroup
6024 %                 \unskip
6025 %                 \hskip -AFTERSKIP \relax %% relax added 14 Jan 91
6026 %                 ELSE \clubpenalty :=G \@clubpenalty
6027 %                 \everypar := NULL
6028 %                 FI
6029 %               }
6030 % FI
6031 %
6032 % END
6033 %
6034 % \@ssect{INDENT}{BEFORESKIP}{AFTERSKIP}{STYLE}{ARG} ==
6035 % BEGIN
6036 % IF AFTERSKIP > 0
6037 % THEN \begingroup
6038 %       STYLE
6039 %       \@hangfrom{\hskip INDENT}{\interlinepenalty 10000 ARG\par}
6040 %       \endgroup
6041 % ELSE \@susechd == BEGIN STYLE
6042 %       \hskip INDENT
6043 %       ARG
6044 %       END
6045 % FI
6046 % \@xsect{AFTERSKIP}
6047 % END
6048 %
6049 % \@afterheading ==
6050 % BEGIN
6051 % @nobreak :=G true
6052 % \everypar := BEGIN IF @nobreak = T
6053 %                 THEN @nobreak :=G false
6054 %                 \clubpenalty :=G 10000
6055 %                 IF @afterindent = F
6056 %                 THEN remove \lastbox
6057 %                 FI
6058 %                 ELSE \clubpenalty :=G \@clubpenalty
6059 %                 \everypar := NULL
6060 %                 FI
6061 %               END
6062 % END
6063 %
6064 % \@secpentalty : The penalty (usually negative) put before a section
6065 %                 heading unless it immediately follows another one.
6066 %

```

```

6067 \newcount\@secpenalty
6068 \@secpenalty = -300
6069
6070
6071 \newif\if@noskipsec \@noskipsectrue
6072
6073
6074 \def\@startsection#1#2#3#4#5#6{\if@noskipsec \leavevmode \fi
6075   \par \@tempkipa #4\relax
6076   \@afterindenttrue
6077   \ifdim \@tempkipa <\z@ \@tempkipa -\@tempkipa \@afterindentfalse\fi
6078   \if@nobreak \everypar{}\else
6079     \addpenalty{\@secpenalty}\addvspace{\@tempkipa}\fi \ifstar
6080     {\@ssect{#3}{#4}{#5}{#6}}{\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}}
6081
6082 \def\@sect#1#2#3#4#5#6[#7]#8{\ifnum #2>\c@secnumdepth
6083   \let\@svsec\@empty\else
6084   \refstepcounter{#1}\edef\@svsec{\csname the#1\endcsname\hskip 1em}\fi
6085   \@tempkipa #5\relax
6086   \ifdim \@tempkipa>\z@
6087     \begingroup #6\relax
6088     \@hangfrom{\hskip #3\relax\@svsec}{\interlinepenalty \@M #8\par}%
6089     \endgroup
6090     \csname #1mark\endcsname{#7}\addcontentsline
6091     {toc}{#1}{\ifnum #2>\c@secnumdepth \else
6092       \protect\numberline{\csname the#1\endcsname}\fi
6093       #7}\else
6094     \def\@svsechd{#6\hskip #3\relax % \relax added 2 May 90
6095       \@svsec #8\csname #1mark\endcsname
6096       #7}\addcontentsline
6097       {toc}{#1}{\ifnum #2>\c@secnumdepth \else
6098         \protect\numberline{\csname the#1\endcsname}\fi
6099         #7}}\fi
6100   \@xsect{#5}}
6101
6102 \def\@xsect#1{\@tempkipa #1\relax
6103   \ifdim \@tempkipa>\z@
6104     \par \nobreak
6105     \vskip \@tempkipa
6106     \@afterheading
6107   \else \global\@nobreakfalse \global\@noskipsectrue
6108     \everypar{\if@noskipsec \global\@noskipsecfalse
6109       \clubpenalty\@M \hskip -\parindent
6110       \begingroup \@svsechd \endgroup \unskip
6111       \hskip -#1\relax % \relax added 14 Jan 91
6112       \else \clubpenalty \@clubpenalty
6113       \everypar{}\fi}\fi\ignorespaces}
6114
6115 \def\@ssect#1#2#3#4#5{\@tempkipa #3\relax
6116   \ifdim \@tempkipa>\z@
6117     \begingroup #4\@hangfrom{\hskip #1}{\interlinepenalty \@M #5\par}\endgroup
6118     \else \def\@svsechd{#4\hskip #1\relax #5}\fi
6119     \@xsect{#3}}
6120
6121 \newif\if@afterindent \@afterindenttrue
6122
6123 \def\@afterheading{\global\@nobreaktrue
6124   \everypar{\if@nobreak
6125     \global\@nobreakfalse
6126     \clubpenalty \@M
6127     \if@afterindent \else {\setbox\z@\lastbox}\fi
6128     \else \clubpenalty \@clubpenalty
6129     \everypar{}\fi}}
6130
6131
6132 % \@hangfrom{TEXT} : Puts TEXT in a box, and makes a hanging indentation
6133 %   of the following material up to the first \par. Should be used
6134 %   in vertical mode.
6135 %
6136 \def\@hangfrom#1{\setbox\@tempboxa\hbox{#1}%
6137   \hangindent \wd\@tempboxa\noindent\box\@tempboxa}
6138
6139 \newcount\c@secnumdepth
6140 \newcount\c@tocdepth

```

```

6141
6142 % \secdef{UNSTARCMSD}{STARCMSD} :
6143 %   When defining a \chapter or \section command without using
6144 %   \startsection, you can use \secdef as follows:
6145 %   \def\chapter { ... \secdef \CMDA \CMDB }
6146 %   \def\CMDA  [#1]#2{ ... } % Command to define \chapter[...]{...}
6147 %   \def\CMDB  #1{ ... }   % Command to define \chapter*{...}
6148
6149 \def\secdef#1#2{\@ifstar{#2}{\@dblarg{#1}}}
6150
6151 % Initializations
6152 %
6153 \def\sectionmark#1{}
6154 \def\subsectionmark#1{}
6155 \def\subsubsectionmark#1{}
6156 \def\paragraphmark#1{}
6157 \def\subparagraphmark#1{}
6158
6159 \message{contents,}
6160 % *****
6161 % *          TABLE OF CONTENTS, ETC.          *
6162 % *****
6163 %
6164 % CONVENTIONS:
6165 % \tf@foo = file number for output for table foo. The file is
6166 %           opened only if @filesw = true.
6167 %
6168 % \contentsline{TYPE}{ENTRY}{PAGE}
6169 %   Macro to produce a TYPE entry in a table of contents, etc.
6170 %   It will appear in the .TOC or other file. For example,
6171 %   The entry for subsection 1.4.3 in the table of contents might
6172 %   be produced by:
6173 %   \contentsline{subsection}{\makebox{30pt}[r]{1.4.3} Gnats and Gnus}{22}
6174 %   The \protect command causes command sequences to be written
6175 %   without expanding them.
6176 %
6177 % \l@TYPE{ENTRY}{PAGE}
6178 %   Macro defined by document style for making an entry of
6179 %   type TYPE in a table of contents, etc. E.g., the document
6180 %   style should define \l@chapter, \l@section, etc.
6181 %
6182 % \addcontentsline{TABLE}{TYPE}{ENTRY}
6183 %   User command for adding his own entry to a table of contents, etc.
6184 %   It adds the entry
6185 %   \contentsline{TYPE}{ENTRY}{page}
6186 %   to the .TABLE file.
6187 %
6188 % \addtocontents{TABLE}{TEXT} : Adds TEXT to the .TABLE file, with no
6189 %   page number.
6190 %
6191 % Note: When used in the ENTRY or TEXT of one of the above commands,
6192 % \protect causes the following control sequence to be written
6193 % on the file without being expanded. The sequence will be expanded
6194 % when the table of contents entry is processed.
6195 %
6196 % SURPRISE: \index, \glossary, and \label are no-ops inside an
6197 % \addcontentsline or \addtocontents command argument. This could cause a
6198 % problem if the user puts an \index or \label into one of the commands he
6199 % writes, or into the optional 'short version' argument of a \section or
6200 % \caption command.
6201 %
6202 % \addcontentsline{TABLE}{TYPE}{ENTRY} ==
6203 % BEGIN
6204 %   if @filesw = true
6205 %   then \begingroup
6206 %         \index == \label == \glossary == \@gobble
6207 %         \protect{ARG} == \string\string\string ARG \string\space\space
6208 %         \@temptokena := \thepage
6209 %         \@tempa == write \string\contentsline
6210 %                   {TYPE}{ENTRY}{\the\@temptokena}
6211 %         \@tempa
6212 %         IF vmode and @nobreak = true THEN \nobreak FI
6213 %   \endgroup
6214 %   fi

```

```

6215 % END
6216 %
6217 % \@starttoc{EXT} : Used to define \tableofcontents, \listoffigures, etc.--
6218 %     e.g., \@starttoc{lof} is used in \listoffigures. This command reads
6219 %     the .EXT file and sets up to write the new .EXT file.
6220 %
6221 % \@starttoc{EXT} ==
6222 % BEGIN
6223 %   \begingroup
6224 %     \makeatletter
6225 %     read_file \jobname.EXT
6226 %     IF @filesw = true
6227 %       THEN open \jobname.EXT as file \tf@EXT
6228 %     FI
6229 %     @nobreak :=G FALSE %% added 24 May 89
6230 %   \endgroup
6231 % END
6232
6233 %% RmS 92/01/14: added \immediate to \openout as all \write commands
6234 %%     are also executed \immediate
6235 \def\@starttoc#1{\begingroup
6236   \makeatletter
6237   \@input{\jobname.#1}\if@filesw \expandafter\newwrite\csname tf@#1\endcsname
6238   \immediate\openout \csname tf@#1\endcsname \jobname.#1\relax
6239   \fi \global\@nobreakfalse \endgroup}
6240
6241
6242 \let\protect=\relax
6243
6244 \def\addcontentsline#1#2#3{\if@filesw \begingroup
6245 \let\label@gobble \let\index@gobble \let\glossary@gobble
6246 \def\protect##1{\string\string\string##1\string\space
6247   \space}\@temptokena{\thepage}%
6248 \edef\@tempa{\write \@auxout{\string\@writefile{#1}{\protect
6249   \contentsline{#2}{#3}{\the\@temptokena}}}\@tempa
6250   \if@nobreak \ifvmode\nobreak\fi\fi\endgroup\fi}
6251
6252 \long\def\addtocontents#1#2{\if@filesw \begingroup
6253 \let\label@gobble \let\index@gobble \let\glossary@gobble
6254 \def\protect##1{\string\string\string##1\string\space\space}%
6255 \edef\@tempa{\write \@auxout {\string\@writefile{#1}{#2}}}\@tempa
6256   \if@nobreak \ifvmode\nobreak\fi\fi\endgroup\fi}
6257
6258 \def\contentsline#1{\csname l@#1\endcsname}
6259
6260 % \@dottedtocline{LEVEL}{INDENT}{NUMWIDTH}{TITLE}{PAGE} :
6261 % Macro to produce a table of contents line with the following
6262 % parameters:
6263 %   LEVEL : If LEVEL > \c@tocdepth, then no line produced.
6264 %   INDENT : Total indentation from the left margin.
6265 %   NUMWIDTH : Width of box for number if the TITLE has a
6266 %               \numberline command.
6267 %               As of 25 Jan 88, this is also the amount of extra indentation
6268 %               added to second and later lines of a multiple line entry.
6269 %   TITLE : Contents of entry.
6270 %   PAGE : Page number.
6271 %
6272 % Uses the following parameters, which must be set by the document style.
6273 % They should be defined with \def's.
6274 %   \@pnumwidth : Width of box in which page number is set.
6275 %   \@tocrmarg : Right margin indentation for all but last line of
6276 %               multiple-line entries.
6277 %   \@dotsep : Separation between dots, in mu units. Should be \def'd to
6278 %               a number like 2 or 1.7
6279 %
6280
6281 %% RmS 91/09/29: added \reset@font for page number
6282 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
6283   \vskip \z@ plus.2\p@
6284   {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
6285     \parindent #2\relax\@afterindenttrue
6286     \interlinepenalty\@M
6287     \leavevmode
6288     \@tempdima #3\relax \advance\leftskip \@tempdima \hbox{} \hskip -\leftskip

```



```

6289 #4\nobreak\leaders\hbox{$\m@th \mkern \@dotsep mu.\mkern \@dotsep
6290 mu$}\hfill \nobreak
6291 \hbox to\@pnumwidth{\hfil\reset@font\rm #5}\par}\fi}
6292
6293
6294 %% Note: \nobreak's added 7 Jan 86 to prevent bad line break that
6295 %% left the page number dangling by itself at left edge of a new line.
6296 %%
6297 %% Changed 25 Jan 88 to use \leftskip instead of \hangindent so
6298 %% leaders of multiple-line contents entries would line up properly.
6299
6300 % \numberline{NUMBER} : For use in a \contentsline command.
6301 % It puts NUMBER flushleft in a box of width \@tempdima
6302 % (Before 25 Jan 88 change, it also added \@tempdima to the hanging
6303 % indentation.)
6304
6305 \def\numberline#1{\hbox to\@tempdima{#1\hfil}}
6306
6307
6308 \message{index,}
6309 %
6310 % ***** INDEX COMMANDS AND GLOSSARY *****
6311 % *****
6312 %
6313 % \makeindex ==
6314 % BEGIN
6315 % if \@filesw = T
6316 % then open file \jobname.IDX as \@indexfile
6317 % \index == BEGIN \@bsphack
6318 % \begingroup
6319 % \protect{X} == \string X\space
6320 % %% added 3 Feb 87 for \index commands
6321 % %% in \footnotes
6322 % re-\catcode special characters to 'other'
6323 % \@wrindex
6324 % fi
6325 % END
6326 %
6327 % \@wrindex{ITEM} ==
6328 % BEGIN
6329 % write of {\indexentry{ITEM}{page number}}
6330 % \endgroup
6331 % \@esphack
6332 % END
6333
6334 % INITIALIZATION:
6335 %
6336 % \index == BEGIN \@bsphack
6337 % \begingroup
6338 % re-\catcode special characters (in case '%' there)
6339 % \@index
6340 % END
6341 %
6342 % \@index{ITEM} == BEGIN \endgroup \@esphack END
6343 %
6344 % Changes made 14 Apr 89 to write \glossaryentry's instead of
6345 % \indexentry's on the .glo file.
6346
6347 \def\makeindex{\if@filesw \newwrite\@indexfile
6348 \immediate\openout\@indexfile=\jobname.idx
6349 \def\index{\@bsphack\begingroup
6350 \def\protect####1{\string####1\space}\@sanitize
6351 \@wrindex}\typeout
6352 {Writing index file \jobname.idx }\fi}
6353
6354 \def\@wrindex#1{\let\thepage\relax
6355 \edef\@tempa{\write\@indexfile{\string
6356 \indexentry{#1}{\thepage}}}\expandafter\endgroup\@tempa
6357 \if\nobreak \ifvmode\nobreak\fi\fi\@esphack}
6358
6359 \def\index{\@bsphack\begingroup \@sanitize\@index}
6360
6361 \def\@index#1{\endgroup\@esphack}
6362

```

```

6363 \def\makeglossary{\if@filesw \newwrite\@glossaryfile
6364 \immediate\openout\@glossaryfile=\jobname.glo
6365 \def\glossary{\@bsphack\begingroup\@sanitize\@wrglossary}\typeout
6366 {Writing glossary file \jobname.glo }\fi}
6367
6368 \def\@wrglossary#1{\let\thepage\relax
6369 \edef\@tempa{\write\@glossaryfile\string
6370 \glossaryentry{#1}{\thepage}}}\expandafter\endgroup\@tempa
6371 \if@nobreak \ifvmode\nobreak\fi\fi\@esphack}
6372
6373 \def\glossary{\@bsphack\begingroup\@sanitize\@index}
6374
6375 \message{bibliography,}
6376 % *****
6377 % * BIBLIOGRAPHY *
6378 % *****
6379 %
6380 % A bibliography is created by the bibliography environment, which
6381 % generates a title such as ‘References’, and a list of entries.
6382 % The BIBTeX program will create a file containing such an environment,
6383 % which will be read in by the \bibliography command. With
6384 % BIBTeX, the following commands will be used.
6385 %
6386 % \bibliography{FILE1,FILE2, ... ,FILEn} : specifies
6387 % the bibdata files. Writes a \bibdata entry on the .aux file
6388 % and tries to read in mainfile.BBL.
6389 %
6390 % \bibliographystyle{STYLE} : Writes a \bibstyle entry on the .aux file.
6391 %
6392 % The thebibliography environment is a list environment. To save the
6393 % use of an extra counter, it should use enumiv as the item counter.
6394 % Instead of using \item, items in the bibliography are produced by the
6395 % following commands:
6396 % \bibitem{NAME} : Produces a numbered entry cited as NAME.
6397 % \bibitem[LABEL]{NAME} : Produces an entry labeled by LABEL and
6398 % cited by NAME.
6399 % The former is used for bibliographies with citations like [1], [2], etc.;
6400 % the latter is used for citations like [Knuth82].
6401 %
6402 % The document style must define the thebibliography environment. This
6403 % environment has a single argument, which is the widest bibliography
6404 % label-- e.g., if the [Knuth67] is the widest entry, then thist argument
6405 % will be Knuth67. The \thebibliography command must begin a list
6406 % environment, which the \endthebibliography command ends.
6407 %
6408 % Entries are cited by the command \cite{NAME}.
6409 %
6410 % PARAMETERS
6411 %
6412 % \@cite : A macro such that \@cite[LABEL1,LABEL2]{NOTE}
6413 % produces the output for a \cite[NOTE]{FOO1,FOO2} command,
6414 % where entry FOOi is defined by \bibitem[LABELi]{FOOi}.
6415 % The switch @tempswa is true if the optional NOTE argument
6416 % is present.
6417 % The default definition is :
6418 % \@cite[LABELS]{NOTE} ==
6419 % BEGIN [LABELS
6420 % IF @tempswa = T THEN , NOTE FI
6421 % ]
6422 % END
6423 %
6424 % \@biblabel : A macro to produce the label in the bibliography
6425 % entry. For \bibitem[LABEL]{NAME}, the label is
6426 % generated by \@biblabel[LABEL]. It has the default
6427 % definition \@biblabel{LABEL} -> [LABEL].
6428 % CONVENTION
6429 %
6430 % \b@FOO : The name or number of the reference created by \cite{FOO}
6431 % E.g., if \cite{FOO} -> [17] , then \b@FOO -> 17.
6432 %
6433 %
6434
6435 \def\bibitem{\@ifnextchar[{\@lbibitem}{\@bibitem}}
6436

```

```

6437 %% RmS 92/02/26: Added \hfill to restore left-alignment of
6438 %%          bibliography labels in alpha style
6439 \def\@lbbibitem[#1]#2{\item[\@biblabel{#1}\hfill]\if@filesw
6440     {\def\protect##1{\string ##1\space}\immediate
6441      \write\@auxout{\string\bibcite{#2}{#1}}}\fi\ignorespaces}
6442 %% Placement of '}' in def of \@lbbibitem corrected 29 Apr 87
6443 %% (Error found by Arthur Ogawa.)
6444
6445 %% RmS 91/11/13: Changed counter enumi to enumiv,
6446 %%          as it says in the comment above
6447 %% RmS 92/01/10: Changed \c@enumiv to \value{\@listctr}.
6448 \def\@bibitem#1{\item\if@filesw \immediate\write\@auxout
6449     {\string\bibcite{#1}{\the\value{\@listctr}}}\fi\ignorespaces}
6450
6451 \def\bibcite#1#2{\global\@namedef{b@#1}{#2}}
6452
6453 \let\citation\@gobble
6454
6455 \def\cite{\@ifnextchar [{\@tempswatrue\@citex}{\@tempswafalse\@citex[]}]%
6456
6457 % \penalty\@m added to definition of \@citex to allow a line
6458 % break after the ',' in citations like [Jones80,Smith77]
6459 % (Added 23 Oct 86)
6460 %
6461 % space added after the ',' (21 Nov 87)
6462 %
6463 %% RmS 91/10/25: added \reset@font, suggested by Bernd Raichle.
6464 %% RmS 91/11/06: added code to remove a leading blank
6465 \def\@citex[#1]#2{\if@filesw\immediate\write\@auxout{\string\citation{#2}}\fi
6466     \let\@citea\@empty
6467     \@citea{\@for\@citeb:=#2\do
6468         {\@citea\def\@citea{,\penalty\@m }%
6469          \def\@tempa##1##2\@nil{\edef\@citeb{\if##1\space##2\else##1##2\fi}}%
6470          \expandafter\@tempa\@citeb\@nil
6471          \ifundefined{b@\@citeb}{\reset@font\bf ?}\@warning
6472           {Citation '\@citeb' on page \thepage \space undefined}}%
6473          \hbox{\csname b@\@citeb\endcsname}}{#1}}
6474
6475 \let\bibdata=\@gobble
6476 \let\bibstyle=\@gobble
6477
6478 \def\bibliography#1{\if@filesw\immediate\write\@auxout{\string\bibdata{#1}}\fi
6479     \@input{\jobname.bbl}}
6480
6481 \def\bibliographystyle#1{\if@filesw\immediate\write\@auxout
6482     {\string\bibstyle{#1}}\fi}
6483
6484 % \nocite{CITATIONS} : puts information on .AUX file to cause
6485 % BibTeX to include the CITATIONS list in the bibliography,
6486 % but puts nothing in the text. (Added 14 Jun 85)
6487
6488 \def\nocite#1{\@bsphack
6489     \if@filesw\immediate\write\@auxout{\string\citation{#1}}\fi
6490     \@esphack}
6491
6492
6493
6494 %DEFAULT DEFINITIONS
6495
6496 \def\@cite#1#2[#{#1\if@tempswa , #2\fi}]
6497 %% RmS 92/01/14: removed \hfill in definition of \@biblabel
6498 \def\@biblabel#1{[#1]}
6499
6500 \message{floats,}
6501 % *****
6502 % *          FLOATS          *
6503 % *****
6504 %
6505 % The different types of floats are identified by a TYPE name, which is
6506 % the name of the counter for that kind of float. For example, figures
6507 % are of type 'figure' and tables are of type 'table'. Each TYPE has
6508 % associated a positive TYPE NUMBER, which is a power of two. E.g.,
6509 % figures might be have type number 1, tables type number 2, programs
6510 % type number 4, etc.

```

```

6511 %
6512 % The locations where a float can go are specified by a PLACEMENT
6513 % SPECIFIER, which is a list of the possible locations, each denoted
6514 % by a letter as follows:
6515 %   h : here - at the current location in the text.
6516 %   t : top - at the top of a text page.
6517 %   b : bottom - at the bottom of a text page.
6518 %   p : page - on a separate float page.
6519 % For example, 'pht' specifies that the float can appear in any of three
6520 % locations: page, here or top.
6521 %
6522 % Where floats may appear on a page, and how many may appear there
6523 % are specified by the following float placement parameters. The
6524 % numbers are named like counters so the user can set them with
6525 % the ordinary counter-setting commands.
6526 %
6527 % \c@topnumber      : Number of floats allowed at the top of a column.
6528 % \topfraction     : Fraction of column that can be devoted to floats.
6529 % \c@dbltopnumber, \dbltopfraction : Same as above, but for double-column
6530 % floats.
6531 % \c@bottomnumber, \bottomfraction : Same as above for bottom of page.
6532 % \c@totalnumber  : Number of floats allowed in a single column,
6533 % including in-text floats.
6534 % \textfraction   : Minimum fraction of column that must contain text.
6535 % \floatpagefraction : Minimum fraction of page that must be taken
6536 % up by float page.
6537 % \dblfloatpagefraction : Same as above, for double-column floats.
6538 %
6539 % The document style must define the following.
6540 %
6541 % \fps@TYPE : The default placement specifier for floats of type TYPE.
6542 %
6543 % \ftype@TYPE : The type number for floats of type TYPE.
6544 %
6545 % \ext@TYPE : The file extension indicating the file on which the
6546 % contents list for float type TYPE is stored. For example,
6547 % \ext@figure = 'lof'.
6548 %
6549 % \fnum@TYPE : A macro to generate the figure number for a caption.
6550 % For example, \fnum@TYPE == Figure \thefigure.
6551 %
6552 % \@makecaption{NUM}{TEXT} : A macro to make a caption, with NUM the value
6553 % produced by \fnum@... and TEXT the text of the caption.
6554 % It can assume it's in a \parbox of the appropriate width.
6555 %
6556 % \@float{TYPE}[PLACEMENT] : This macro begins a float environment for a
6557 % single-column float of type TYPE with PLACEMENT as the placement
6558 % specifier. The default value of PLACEMENT is defined by \fps@TYPE.
6559 % The environment is ended by \end@float.
6560 % E.g., \figure == \@float{figure}, \endfigure == \end@float.
6561 %
6562 % \caption ==
6563 % BEGIN
6564 %   \refstepcounter{\@capttype}
6565 %   \@dblarg{\@caption{\@capttype}}
6566 % END
6567 %
6568 %% In following definition, \par moved from after \addcontentsline to
6569 %% before \addcontentsline because the \write could cause
6570 %% an extra blank line to be added to the paragraph above the
6571 %% caption. (Change made 12 Jun 87)
6572 %
6573 % \@caption{TYPE}[STEXT]{TEXT} ==
6574 % BEGIN
6575 %   \par
6576 %   \addcontentsline{\ext@TYPE}{TYPE}{\numberline{\theTYPE}{STEXT}}
6577 %   \begingroup
6578 %     \@parboxrestore
6579 %     \normalsize
6580 %     \@makecaption{\fnum@TYPE}{TEXT}
6581 %   \par
6582 %   \endgroup
6583 % END
6584 %

```

```

6585 % \@float{TYPE}[PLACEMENT] ==
6586 % BEGIN
6587 %   if hmode then \@sphack
6588 %       \@floatpenalty := -10002
6589 %       else \@floatpenalty := -10003
6590 %   fi
6591 %   \@capttype ==L TYPE
6592 %   if inner
6593 %       then LaTeX Error: 'Not in outer paragraph mode.'
6594 %       \@floatpenalty := 0
6595 %       else if \@freelist nonempty
6596 %           then \@currbox :=L head of \@freelist
6597 %               \@freelist :=G tail of \@freelist
6598 %               \count\@currbox :=G 32*\ftype@TYPE + 16 +
6599 %                   bits determined by PLACEMENT
6600 %           else \@floatpenalty := 0
6601 %               LaTeX Error: 'Too many unprocessed floats'
6602 %       fi
6603 %   fi
6604 %   \@currbox :=G \vbox{%% 15 Dec 87 -- removed \boxmaxdepth :=L Opt
6605 %               %% that made box zero depth because it screwed
6606 %               %% things up. Instead, added \vskip Opt at end
6607 %               \hsize = \columnwidth
6608 %               \@parboxrestore
6609 % END
6610 %
6611 % \end@float ==
6612 % BEGIN
6613 %   \vskip Opt %% makes 0 depth box -- added 15 Dec 87
6614 % }
6615 %   if \@floatpenalty < 0
6616 %       then add \@currbox to \@currlist
6617 %       if \ht\@currbox > \textheight
6618 %       then \ht\@currbox :=G \textheight fi
6619 %       if \@floatpenalty < -10002
6620 %       then \penalty -10004
6621 %           \vbox{}
6622 %           \penalty \@floatpenalty
6623 %       else \vadjust{\penalty -10004
6624 %           \vbox{}
6625 %           \penalty \@floatpenalty}
6626 %       \@Esphack
6627 %   fi   fi
6628 % END
6629 %
6630 % \@dblfloat{TYPE}[PLACEMENT] : Macro to begin a float environment for a
6631 % double-column float of type TYPE with PLACEMENT as the placement
6632 % specifier. The default value of PLACEMENT is 'tp'
6633 % The environment is ended by \enddblfloat.
6634 % E.g., \figure* == \@dblfloat{figure}, \endfigure* == \enddblfloat.
6635 %
6636 % \@dblfloat{TYPE}[PLACEMENT] ==
6637 % Identical to \@float{TYPE}[PLACEMENT] except \hsize and \linewidth
6638 % are set to \textwidth.
6639 %
6640 % \enddblfloat ==
6641 % BEGIN   %% { BRACE MATCHING
6642 %   \vskip Opt %% makes 0 depth box -- added 15 Dec 87
6643 % }
6644 %   if \@floatpenalty < 0
6645 %       then \@dbldeferlist :=G \@dbldeferlist * \@currbox
6646 %   fi
6647 %   if \@floatpenalty = -10002 then \@Esphack fi
6648 % END
6649 %
6650 \newcount\@floatpenalty
6651
6652 \def\caption{\refstepcounter\@capttype \@dblarg{\@caption\@capttype}}
6653
6654
6655 \long\def\@caption#1[#2]#3{\par\addcontentsline{\csname
6656   ext@#1\endcsname}{#1}{\protect\numberline{\csname
6657   the#1\endcsname}{\ignorespaces #2}}\begingroup
6658   \@parboxrestore

```

```

6659 \normalsize
6660 \makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
6661 \endgroup}
6662
6663 \def\@float#1{\@ifnextchar[{\@xfloat{#1}}{\edef\@tempa{\noexpand\@xfloat
6664 {#1}[\csname fps@#1\endcsname]}\@tempa}}
6665
6666 \def\@xfloat#1[#2]{\ifhmode \@bsphack\@floatpenalty -\@Mii\else
6667 \@floatpenalty-\@Miii\fi\def\@capttype{#1}\ifinner
6668 \@parmoderr\@floatpenalty\z@
6669 \else\@next\@currbox\@freelist{\@tempcnta\csname ftype@#1\endcsname
6670 \multiply\@tempcnta\@xxxiii\advance\@tempcnta\sixt@@n
6671 \@tfor \@tempa :=#2\do
6672 {\if\@tempa h\advance\@tempcnta \@ne\fi
6673 \if\@tempa t\advance\@tempcnta \tw@\fi
6674 \if\@tempa b\advance\@tempcnta 4\relax\fi
6675 \if\@tempa p\advance\@tempcnta 8\relax\fi
6676 }\global\count\@currbox\@tempcnta}\@fltovf\fi
6677 \global\setbox\@currbox\ vbox\bgroup
6678 % \boxmaxdepth\z@ % commented out 15 Dec 87
6679 \hsize\columnwidth \@parboxrestore}
6680
6681 \def\end@float{\par\vskip\z@\egroup %% \par\vskip\z@ added 15 Dec 87
6682 \ifnum\@floatpenalty <\z@
6683 \@cons\@currlist\@currbox
6684 \ifdim \ht\@currbox >\textheight
6685 %% RmS 91/11/06 added warning message
6686 % perhaps we should use an error message
6687 \warning{Float larger than \string\textheight}%
6688 \ht\@currbox\textheight \fi
6689 \ifnum\@floatpenalty <-\@Mii
6690 \penalty -\@Miv
6691 \@tempdima\prevdepth %% saving and restoring \prevdepth added
6692 \vbox{} %% 26 May 87 to prevent extra vertical
6693 \prevdepth \@tempdima %% space when used in vertical mode
6694 \penalty\@floatpenalty
6695 %% RmS 92/03/18 changed \@esphack to \@Esphack
6696 \else \vadjust{\penalty -\@Miv \vbox{} \penalty\@floatpenalty}\@Esphack
6697 \fi\fi}
6698
6699
6700 \def\@dblfloat{\if@twocolumn\let\@tempa\@dblft\else\let\@tempa\@float\fi
6701 \@tempa}
6702
6703 \def\@dblft#1{\@ifnextchar[{\@xdblfloat{#1}}{\@xdblfloat{#1}[tp]}}
6704
6705 \def\@xdblfloat#1[#2]{\@xfloat{#1}[#2]\hsize\textwidth\linewidth\textwidth}
6706
6707 \def\end@dblfloat{\if@twocolumn
6708 \par\vskip\z@\egroup %% \par\vskip\z@ added 15 Dec 87\egroup
6709 \ifnum\@floatpenalty <\z@
6710 % make sure that we never exceed \textheight, otherwise float
6711 % will never get typeset =91/03/15 FMi=
6712 \ifdim\ht\@currbox >\textheight
6713 % perhaps we should use an error message
6714 \warning{Float larger than \string\textheight}%
6715 \ht\@currbox\textheight \fi
6716 \@cons\@dbldeferlist\@currbox\fi
6717 %% RmS 92/03/18 changed \@esphack to \@Esphack
6718 \ifnum \@floatpenalty =-\@Mii \@Esphack\fi\else\end@float\fi}
6719
6720 \newcount\c@topnumber
6721 \newcount\c@dbltopnumber
6722 \newcount\c@bottomnumber
6723 \newcount\c@totalnumber
6724
6725 \def\@floatplacement{\global\@topnum\c@topnumber
6726 \global\@toproom \topfraction\@colht
6727 \global\@botnum \c@bottomnumber
6728 \global\@botroom \bottomfraction\@colht
6729 \global\@colnum \c@totalnumber
6730 \@fpmin \floatpagefraction\@colht}
6731
6732 \def\@dblfloatplacement{\global\@dbltopnum\c@dbltopnumber

```

```

6733 \global\@dbltoproom \dbltopfraction\@colht
6734 \@fpmmin \dblfloatpagefraction\textheight
6735 \fptop \dblfpptop
6736 \fpsep \dblfpsep
6737 \fpbot \dblfpbot}
6738
6739 % MARGINAL NOTES:
6740 %
6741 % Marginal notes use the same mechanism as floats to communicate
6742 % with the \output routine. Marginal notes are distinguished from
6743 % floats by having a negative placement specification. The command
6744 % \marginpar [LTEXT]{RTEXT} generates a marginal note in a parbox,
6745 % using LTEXT if it's on the left and RTEXT if it's on the right.
6746 % (Default is RTEXT = LTEXT.) It uses the following parameters.
6747 %
6748 % \marginparwidth : Width of marginal notes.
6749 % \marginparsep : Distance between marginal note and text.
6750 % the page layout to determine how to move the marginal
6751 % note into the margin. E.g., \@leftmargin skip ==
6752 % \hskip -\marginparwidth \hskip -\marginparsep .
6753 % \marginparpush : Minimum vertical separation between \marginpar's
6754 %
6755 % Marginal notes are normally put on the outside of the page
6756 % if @mparswitch = true, and on the right if @mparswitch = false.
6757 % The command \reversemarginpar reverses the side where they
6758 % are put. \normalmarginpar undoes \reversemarginpar.
6759 % These commands have no effect for two-column output.
6760 %
6761 % SURPRISE: if two marginal notes appear on the same line of
6762 % text, then the second one could appear on the next page, in
6763 % a funny position.
6764 %
6765 %
6766 % \marginpar [LTEXT]{RTEXT} ==
6767 % BEGIN
6768 % if hmode then \bsphack
6769 % \floatpenalty := -10002
6770 % else \floatpenalty := -10003
6771 % fi
6772 % if inner
6773 % then LaTeX Error: 'Not in outer paragraph mode.'
6774 % \floatpenalty := 0
6775 % else if \@freelist has two elements:
6776 % then get \@marbox, \@currbox from \@freelist
6777 % \count\@marbox :=G -1
6778 % else \floatpenalty := 0
6779 % LaTeX Error: 'Too many unprocessed floats'
6780 % \@currbox, \@marbox := \@tempboxa %%use \def
6781 % fi
6782 % fi
6783 % if optional argument
6784 % then %% \@xmpar ==
6785 % \savemarbox\@marbox{LTEXT}
6786 % \savemarbox\@currbox{RTEXT}
6787 % else %% \@ympar ==
6788 % \savemarbox\@marbox{RTEXT}
6789 % \box\@currbox :=G \box\@marbox
6790 % fi
6791 % %% \@xympar ==
6792 % if \floatpenalty < 0 then add \@marbox to \@currlist fi
6793 % \setbox\@tempboxa =L %% added 3 Jan 88 to correct bug introduced
6794 % { \end@float %%%% BRACE MATCHING} %% by 15 Dec 87 change
6795 % END
6796 %
6797 % \savemarbox\BOX{TEXT} ==
6798 % BEGIN
6799 % \BOX :=G \vtop{ \hspace = \marginparwidth
6800 % \parbox\@restore
6801 % TEXT
6802 % }
6803 % END
6804 %
6805 % \reversemarginpar == BEGIN \@mparbottom :=G 0
6806 % @reversemargin :=G true

```

```

6807 %                               END
6808 %
6809 % \normalmarginpar == BEGIN \mparbottom :=G 0
6810 %                               @reversemargin :=G false
6811 %                               END
6812 %
6813 %
6814 \def\marginpar{\ifhmode \bsphack\floatpenalty -\@Mii\else
6815 \floatpenalty-\@Miii\fi\ifinner
6816 \parmoderr\floatpenalty\z@
6817 \else\@next\currbox\freelist{}-\@next\marbox\freelist{\global
6818 \count\marbox\m@ne}\floatpenalty\z@ \fltovf\def\currbox{\tempboxa
6819 }\def\marbox{\tempboxa}\fi
6820 \ifnextchar [{\xmpar}{\ympar}}
6821 %
6822 \long\def\xmpar[#1]#2{\savemarbox\marbox{#1}\savemarbox\currbox
6823 {#2}\xypar}
6824 %
6825 \long\def\ympar#1{\savemarbox\marbox{#1}\global\setbox\currbox
6826 \copy\marbox\xypar}
6827 %
6828 \long\def@savemarbox#1#2{\global\setbox#1\top{\hsize\marginparwidth
6829 \parboxrestore #2}}
6830 %
6831 \def\xypar{\ifnum\floatpenalty <\z@\@cons\currlist\marbox\fi
6832 \setbox\tempboxa\ vbox %% added 3 Jan 88
6833 %% RmS 92/03/18 added \global\ignorefalse
6834 \bgroup\endfloat\global\ignorefalse\esphack}
6835 %
6836 \def\reversemarginpar{\global\mparbottom\z@ \reversemargintrue}
6837 \def\normalmarginpar{\global\mparbottom\z@ \reversemarginfalse}
6838 %
6839 %
6840 \message{footnotes,}
6841 % *****
6842 % * FOOTNOTES *
6843 % *****
6844 %
6845 % \footnote{NOTE} : User command to insert a footnote.
6846 %
6847 % \footnote[NUM]{NOTE} : User command to insert a footnote numbered
6848 % NUM, where NUM is a number -- 1, 2,
6849 % etc. For example, if footnotes are numbered
6850 % *, **, etc. within pages, then \footnote[2]{...}
6851 % produces footnote '*'. This command does not
6852 % step the footnote counter.
6853 %
6854 % \footnotemark[NUM] : Command to produce just the footnote mark in
6855 % the text, but no footnote. With no argument,
6856 % it steps the footnote counter before generating
6857 % the mark.
6858 %
6859 % \footnotetext[NUM]{TEXT} : Command to produce the footnote but no
6860 % mark. \footnote is equivalent to
6861 % \footnotemark \footnotetext .
6862 %
6863 % As in PLAIN, footnotes use \insert\footins, and the following parameters:
6864 %
6865 % \footnotesize : Size-changing command for footnotes.
6866 %
6867 % \footnotesep : The height of a strut placed at the beginning of
6868 % every footnote.
6869 % \skip\footins : Space between main text and footnotes. The rule
6870 % separating footnotes from text occurs in this space.
6871 % This space lies above the strut of height \footnotesep
6872 % which is at the beginning of the first footnote.
6873 % \footnoterule : Macro to draw the rule separating footnotes from text.
6874 % It is executed right after a \vspace of \skip\footins.
6875 % It should take zero vertical space--i.e., it should to
6876 % a negative skip to compensate for any positive space
6877 % it occupies. (See PLAIN.TEX.)
6878 %
6879 % \interfootnotelinepenalty : Interline penalty for footnotes.
6880 %

```



```

6881 % \thefootnote : In usual LaTeX style, produces the footnote number.
6882 %           If footnotes are to be numbered within pages, then the
6883 %           document style file must include an \addtoreset command
6884 %           to cause the footnote counter to be reset when the page
6885 %           counter is stepped. This is not a good idea, though,
6886 %           because the counter will not always be reset in time
6887 %           to ensure that the first footnote on a page is footnote
6888 %           number one.
6889 %
6890 % \@thefnmark : Holds the current footnote's mark--e.g., \dag or '1' or 'a'.
6891 %
6892 % \@mpfnnumber : A macro that generates the numbers for \footnote
6893 %               and \footnotemark commands. It == \thefootnote
6894 %               outside a minipage environment, but can be changed
6895 %               inside to generate numbers for \footnote's.
6896 %
6897 % \@makefnmark : A macro to generate the footnote marker from \@thefnmark
6898 %               The default definition is \hbox{ $\hat{\}$ \@thefnmark$}.
6899 %
6900 % \@makefntext{NOTE} :
6901 %           Must produce the actual footnote, using \@thefnmark as the mark
6902 %           of the footnote and NOTE as the text. It is called when effectively
6903 %           inside a \parbox, with \hsize = \columnwidth. For example, it might
6904 %           be as simple as
6905 %           $\hat{\@thefnmark}$ NOTE
6906 %
6907 % In a minipage environment, \footnote and \footnotetext are redefined
6908 % so that
6909 % (a) they use the counter mpfootnote
6910 % (b) the footnotes they produce go at the bottom of the minipage.
6911 % The switch is accomplished by letting \@mpfn == footnote or mpfootnote
6912 % and \themPFN == \thefootnote or \themPFN, and by redefining
6913 % \@footnotetext to be \@mpfootnotetext in the minipage.
6914 %
6915 % \footnote{NOTE} ==
6916 % BEGIN
6917 %   \stepcounter{\@mpfn}
6918 %   \@thefnmark :=G eval (\themPFN)
6919 %   \@footnotemark
6920 %   \@footnotetext{NOTE}
6921 % END
6922 %
6923 % \footnote[NUM]{NOTE} ==
6924 % BEGIN
6925 %   begingroup
6926 %     counter \@mpfn :=L NUM
6927 %     \@thefnmark :=G eval (\themPFN)
6928 %   endgroup
6929 %   \@footnotemark
6930 %   \@footnotetext{NOTE}
6931 % END
6932 %
6933 % \@footnotetext{NOTE} ==
6934 % BEGIN
6935 %   \insert into \footins
6936 %     {\footnotesize
6937 %       \interlinepenalty :=L \interfootnotelinepenalty
6938 %       \splittopskip :=L \footnotesep
6939 %       \splitmaxdepth :=L \dp\strutbox
6940 %       \floatingpenalty :=L 20000
6941 %       \hsize :=L \columnwidth
6942 %       \@parboxrestore
6943 %       set \@currentlabel to make \label command work right
6944 %       \@makefntext{\rule{Opt}{\footnotesep} NOTE}
6945 %     }
6946 % END
6947 %
6948 % \footnotemark ==
6949 % BEGIN \stepcounter{footnote}
6950 %   \@thefnmark :=G eval(\thefootnote)
6951 %   \@footnotemark
6952 % END
6953 %
6954 % \footnotemark[NUM] ==

```

```

6955 % BEGIN
6956 %   begingroup
6957 %     footnote counter :=L NUM
6958 %     \@thefnmark :=G eval(\thefootnote)
6959 %   endgroup
6960 %   \@footnotemark
6961 % END
6962 %
6963 % \@footnotemark ==
6964 % BEGIN
6965 %   \leavevmode
6966 %   IF hmode THEN \@xsf := \the\spacefactor FI
6967 %   \@makefnmark      % put number in main text
6968 %   IF hmode THEN \spacefactor := \@xsf FI
6969 % END
6970 %
6971 % \footnotetext ==
6972 % BEGIN \@thefnmark :=G eval (\thempfn)
6973 %   \@footnotetext
6974 % END
6975 %
6976 % \footnotetext[NUM] ==
6977 % BEGIN begingroup counter \@mpfn :=L NUM
6978 %   \@thefnmark :=G eval (\thempfn)
6979 %   endgroup
6980 %   \@footnotetext
6981 % END
6982 %
6983
6984 \@definecounter{footnote}
6985 \def\thefootnote{\arabic{footnote}}
6986
6987 \@definecounter{mpfootnote}
6988 \def\thempfootnote{\alph{mpfootnote}}
6989
6990 % Default definition
6991 \def\@makefnmark{\hbox{$\sim\@thefnmark\m@th$}}
6992
6993 \newdimen\footnotesep
6994
6995 %% RmS 91/11/01: Added \let\protect\noexpand in \footnote, \footnotemark,
6996 %%               and \footnotetext, since \xdef is used.
6997 %% RmS 91/11/22: Added \let\protect\noexpand in \@xfootnote, \@xfootnotemark,
6998 %%               and \@xfootnotetext.
6999
7000 \def\footnote{\@ifnextchar[{\@xfootnote}{\stepcounter{\@mpfn}}%
7001   \begingroup\let\protect\noexpand
7002   \xdef\@thefnmark{\thempfn}\endgroup
7003   \@footnotemark\@footnotetext}}
7004
7005 \def\@xfootnote[#1]{\begingroup \csname c@\mpfn\endcsname #1\relax
7006   \let\protect\noexpand
7007   \xdef\@thefnmark{\thempfn}\endgroup
7008   \@footnotemark\@footnotetext}
7009
7010 %% RmS 91/09/29: added \reset@font
7011 \long\def\@footnotetext#1{\insert\footins{\reset@font\footnotesize
7012   \interlinepenalty\interfootnotelinepenalty
7013   \splittopskip\footnotesep
7014   \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
7015   \hsize\columnwidth \@parboxrestore
7016   \edef\@currentlabel{\csname p@footnote\endcsname\@thefnmark}\@makefnmark
7017   {\rule{\z@}{\footnotesep}}\ignorespaces
7018   #1\strut}}
7019
7020 \def\footnotemark{\@ifnextchar[{\@xfootnotemark}{\stepcounter{footnote}}%
7021   \begingroup\let\protect\noexpand
7022   \xdef\@thefnmark{\thefootnote}\endgroup
7023   \@footnotemark}}
7024
7025 \def\@xfootnotemark[#1]{\begingroup \c@footnote #1\relax
7026   \let\protect\noexpand
7027   \xdef\@thefnmark{\thefootnote}\endgroup \@footnotemark}
7028

```

```

7029 \def\@footnotemark{\leavevmode\ifhmode
7030 \edef\x@sf{\the\spacefactor}\fi \@makefnmark
7031 \ifhmode\spacefactor\x@sf\fi\relax}
7032
7033 \def\footnotetext{\@ifnextchar [{\@xfootnotenext}]%
7034 {\begingroup\let\protect\noexpand
7035 \xdef\@thefnmark{\thempfn}\endgroup
7036 \@footnotetext}}
7037
7038 \def\@xfootnotenext[#1]{\begingroup \csname c@\@mpfn\endcsname #1\relax
7039 \let\protect\noexpand
7040 \xdef\@thefnmark{\thempfn}\endgroup \@footnotetext}
7041
7042 \def\@mpfn{footnote}
7043 \def\thempfn{\thefootnote}
7044
7045 \message{initial,}
7046 %
7047 % *****
7048 % * INITIAL DECLARATION COMMANDS *
7049 % *****
7050 %
7051 % DOCUMENT STYLE
7052 % -----
7053 % The user starts his file with the command
7054 % \documentstyle [OPTION1, ... ,OPTIONk]{STYLE}
7055 % which saves the OPTION's and \input's the file STYLE.STY. When the
7056 % STYLE.STY file issues the command \options, the following happens
7057 % for each i :
7058 % IF \ds@OPTIONi is defined
7059 % THEN execute \ds@OPTIONi
7060 % ELSE save OPTIONi on a list of unprocessed options.
7061 % FI
7062 % After STYLE.STY has been executed, the file OPTIONi.STY is read for
7063 % each OPTIONi on the list of unprocessed options.
7064 %
7065 % \documentstyle ==
7066 % BEGIN
7067 % IF next char = [
7068 % THEN \documentstyle
7069 % ELSE \@documentstyle[]
7070 % FI
7071 % END
7072 %
7073 % \@documentstyle[OPTIONS]{STYLE} ==
7074 % BEGIN
7075 % \makeatletter
7076 % \@optionlist := OPTIONS
7077 % \@optionfiles :=G null
7078 % \input STYLE.STY
7079 % \@elt == \input
7080 % \@optionfiles
7081 % \@elt == \relax
7082 % \makeatother
7083 % END
7084 %
7085 % \@options ==
7086 % BEGIN
7087 % \@elt := \relax
7088 % FOR \@tempa := \@optionlist
7089 % DO IF \ds@[eval(\@tempa)] defined
7090 % THEN \ds@[eval(\@tempa)]
7091 % ELSE \@optionfiles :=G \@optionfiles *
7092 % \@elt eval(\@tempa) \relax
7093 % OD FI
7094 % END
7095 %
7096 % PAGE STYLE COMMANDS
7097 % -----
7098 % \pagestyle{STYLE} : sets the page style of the current and succeeding
7099 % pages to STYLE
7100 %
7101 % \thispagestyle{STYLE} : sets the page style of the current page only
7102 % to STYLE

```

```

7103 %
7104 % To define a page style STYLE, you must define \ps@STYLE to set the page
7105 % style parameters.
7106 %
7107 % HOW A PAGE STYLE MAKES RUNNING HEADS AND FEET:
7108 %
7109 % The \ps@... command defines the macros \@oddhead, \@oddfoot,
7110 % \@evenhead, and \@evenfoot to define the running heads and feet.
7111 % (See output routine.) To make headings determined by the sectioning
7112 % commands, the page style defines the commands \chaptermark,
7113 % \sectionmark, etc., where \chaptermark{TEXT} is called by \chapter to
7114 % set a mark. The \...mark commands and the \...head macros are defined
7115 % with the help of the following macros. (All the \...mark commands
7116 % should be initialized to no-ops.)
7117 %
7118 % MARKING CONVENTIONS:
7119 % LaTeX extends TeX's \mark facility by producing two kinds of marks
7120 % a 'left' and a 'right' mark, using the following commands:
7121 % \markboth{LEFT}{RIGHT} : Adds both marks.
7122 % \markright{RIGHT} : Adds a 'right' mark.
7123 % \leftmark : Used in the output routine, gets the current 'left' mark.
7124 % Works like TeX's \botmark.
7125 % \rightmark : Used in the output routine, gets the current 'right' mark.
7126 % Works like TeX's \firstmark.
7127 % The marking commands work reasonably well for right marks 'numbered
7128 % within' left marks--e.g., the left mark is changed by a \chapter command and
7129 % the right mark is changed by a \section command. However, it does
7130 % produce somewhat anomalous results if 2 \markboth's occur on the same page.
7131 %
7132 % Commands like \tableofcontents that should set the marks in some page styles
7133 % use a \mkboth command, which is \let by the pagestyle command (\ps@...)
7134 % to \markboth for setting the heading or to \gobbletwo to do nothing.
7135 %
7136 \def\documentstyle{\@ifnextchar[{\@documentstyle}{\@documentstyle []}}
7137 %
7138 \def\@documentstyle[#1]#2{\makeatletter
7139 \def\@optionlist{#1}\gdef\@optionfiles{\input #2.sty\relax
7140 \let\@elt\input \@optionfiles \let\@elt\relax \makeatother}
7141 %
7142 \def\@options{\let\@elt\relax
7143 \@for\@tempa:=\@optionlist\do
7144 {\@ifundefined{ds@\@tempa}{\xdef\@optionfiles{\@optionfiles
7145 \@elt \@tempa.sty\relax}}{\csname ds@\@tempa\endcsname}}}
7146 %
7147 \def\pagestyle#1{\@nameuse{ps@#1}}
7148 \def\thispagestyle#1{\global\@specialpagetrue\gdef\@specialstyle{#1}}
7149 %
7150 % \head : An obsolete command that was used in the 'myheadings'
7151 % page style. (Removed 14 Jun 85)
7152 % \def\head{\@ifnextchar[{\@thead}{\@yhead}}
7153 % \def\@thead[#1]#2{\if #1l \def\@lhead{#2}\else \def\@rhead{#2}\fi}
7154 % \def\@yhead#1{\def\@lhead{#1}\def\@rhead{#1}}
7155 %
7156 % Initialization
7157 %
7158 %\def\@lhead{} %% RmS 91/09/29: removed since no longer used
7159 %\def\@rhead{} %% ditto
7160 %
7161 %
7162 % Default Initializations
7163 %
7164 \def\ps@empty{\let\mkboth\gobbletwo\let\@oddhead\empty\let\@oddfoot\empty
7165 \let\@evenhead\empty\let\@evenfoot\empty}
7166 %
7167 \def\ps@plain{\let\mkboth\gobbletwo
7168 \let\@oddhead\empty\def\@oddfoot{\reset\font\rm\hfil\thepage
7169 \hfil}\let\@evenhead\empty\let\@evenfoot\@oddfoot}
7170 %
7171 \def\@leftmark#1#2{#1}
7172 \def\@rightmark#1#2{#2}
7173 %
7174 %% test for @nobreak added 15 Apr 86 in \markboth and \markright
7175 %% letting \label and \index to \relax added 22 Feb 86 so these
7176 %% commands can appear in sectioning command arguments

```

```

7177 %% RmS 91/06/21 Same for \glossary
7178 %%
7179
7180 \def\markboth#1#2{\gdef\@themark{#{1}#{2}}{\let\protect\noexpand
7181 \let\label\relax \let\index\relax \let\glossary\relax
7182 \mark{\@themark}}\if@nobreak\ifvmode\nobreak\fi\fi}
7183 \def\markright#1{\let\protect\noexpand
7184 \let\label\relax \let\index\relax \let\glossary\relax
7185 \expandafter\@markright\@themark
7186 {#1}\mark{\@themark}}\if@nobreak\ifvmode\nobreak\fi\fi}
7187
7188 \def\@markright#1#2#3{\gdef\@themark{#{1}#{3}}}
7189 \def\leftmark{\expandafter\@leftmark\botmark}
7190 \def\rightmark{\expandafter\@rightmark\firstmark}
7191
7192 % Initialization
7193 %
7194 \def\@themark{#{}}
7195
7196
7197 % OTHER
7198 % -----
7199 %
7200 % \raggedbottom : Typesets pages with no vertical stretch, so they have
7201 % their natural height instead of all being exactly the
7202 % same height. (Uses a space of .0001fil to avoid
7203 % interfering with the ifil space of \newpage.)
7204 %
7205 % \flushbottom : Inverse of \raggedbottom - makes all pages the same
7206 % height.
7207 %
7208 % \sloppy : Resets TeX's parameters so it accepts worse line and page
7209 % breaks, and slightly more overfull boxes.
7210 %
7211 % \fussy : Resets TeX's parameters to their normal finnickly values.
7212 %
7213
7214 \def\raggedbottom{\def\@textbottom{\vskip \z@ plus.0001fil}\let\@texttop\relax}
7215 \def\flushbottom{\let\@textbottom\relax \let\@texttop\relax}
7216
7217 % Default definitions
7218 % \sloppy will never (well, hardly ever) produce overfull boxes, but may
7219 % produce underfull ones. (14 June 85)
7220 % A sloppypar environment is equivalent to {\par \sloppy ... \par}.
7221 \def\sloppy{\tolerance \M \hfuzz .5\p@ \vfuzz .5\p@}
7222 \def\sloppypar{\par\sloppy}
7223 \def\endsloppypar{\par}
7224 \def\fussy{\tolerance 200 \hfuzz .1\p@ \vfuzz .1\p@}
7225
7226
7227
7228 % LaTeX default is no overfull box rule. Changed by document
7229 % style option
7230
7231 \overfullrule 0pt
7232
7233 \message{output,}
7234 % *****
7235 % * OUTPUT *
7236 % *****
7237 %
7238 %
7239 % PAGE LAYOUT PARAMETERS
7240 %
7241 % \topmargin : Extra space added to top of page.
7242 % @twoside : boolean. T if two-sided printing
7243 % \oddsidemargin : IF @twoside = T
7244 % THEN extra space added to left of odd-numbered
7245 % pages.
7246 % ELSE extra space added to left of all pages.
7247 % \evensidemargin : IF @twoside = T
7248 % THEN extra space added to left of even-numbered
7249 % pages.
7250 % \headheight : height of head

```

```

7251 % \headsep      : separation between head and text
7252 % \footskip     : distance separation between baseline of last
7253 %               : line of text and baseline of foot.
7254 %               : Note difference between \footSKIP and \headSEP.
7255 % \textheight  : height of text on page, excluding head and foot
7256 % \textwidth   : width of printing on page
7257 % \columnsep   : IF @twocolumn = T
7258 %               : THEN width of space between columns
7259 % \columnseprule : IF @twocolumn = T
7260 %               : THEN width of rule between columns (0 if none).
7261 % \columnwidth : IF @twocolumn = T
7262 %               : THEN (\textwidth - \columnsep)/2
7263 %               : ELSE \textwidth
7264 %               : It is set by the \@maketwocolumn and \@makeonecolumn
7265 %               : commands.
7266 % \@textbottom : Command executed at bottom of vbox holding text of page
7267 %               : (including figures). The \raggedbottom command
7268 %               : almost \let's this to \vfil (actually sets it to
7269 %               : \vskip \z@ plus.0001fil). %expanded 18 Jun 86
7270 %
7271 % \@texttop    : Command executed at top of vbox holding text of page
7272 %               : (including figures). Used by letter style; can also
7273 %               : be used to produce centered pages. Is \let to \relax
7274 %               : by \raggedbottom and \flushbottom.
7275 %
7276 % Page layout must also initialize \@colht and \@colroom to \textheight.
7277 %
7278 % PAGE STYLE PARAMETERS:
7279 %
7280 % \floatsep     : Space left between floats.
7281 % \textfloatsep : Space between last top float or first bottom float
7282 %               : and the text.
7283 % \topfigrule   : Command to place rule (or whatever) between floats
7284 %               : at top of page and text. Executed in inner vertical
7285 %               : mode right before the \textfloatsep skip separating
7286 %               : the floats from the text. Must occupy zero vertical
7287 %               : space. (See \footnoterule.)
7288 % \botfigrule   : Same as \topfigrule, but put after the \textfloatsep
7289 %               : skip separating text from the floats at bottom of page.
7290 % \intextsep    : Space left on top and bottom of an in-text float.
7291 % \@maxsep     : The maximum of \floatsep, \textfloatsep and \intextsep
7292 % \dblfloatsep : Space between double-column floats.
7293 % \dbltextfloatsep : Space between top or bottom double-column floats
7294 %               : and text.
7295 % \dblfigrule   : Similar to \topfigrule, but for double-column floats.
7296 % \@dblmaxsep  : The maximum of \dblfloatsep and \dbltextfloatsep
7297 % \@fptop     : Glue to go at top of float column -- must be Opt +
7298 %               : stretch
7299 % \@fpsep     : Glue to go between floats in a float column.
7300 % \@fpbot     : Glue to go at bottom of float column -- must be Opt +
7301 %               : stretch
7302 % \@dblfpsep, \@dblfpbot : Analogous for double-column float page in two-column
7303 %               : format.
7304 %
7305 %
7306 % FOOTNOTES: As in PLAIN, footnotes use \insert\footins.
7307 %
7308 % PAGE LAYOUT SWITCHES AND MACROS
7309 %
7310 % @twocolumn    : Boolean. T if two columns per page.
7311 %
7312 % PAGE STYLE MACROS AND SWITCHES
7313 %
7314 % \@oddhead     : IF @twoside = T
7315 %               : THEN macro to generate head of odd-numbered pages.
7316 %               : ELSE macro to generate head of all pages.
7317 % \@evenhead    : IF @twoside = T
7318 %               : THEN macro to generate head of even-numbered pages.
7319 % \@oddfoot     : IF @twoside = T
7320 %               : THEN macro to generate foot of odd-numbered pages.
7321 %               : ELSE macro to generate foot of all pages.
7322 % \@evenfoot    : IF @twoside = T
7323 %               : THEN macro to generate foot of even-numbered pages.
7324 % @specialpage : boolean. T if current page is to have a special format.

```

```

7325 % \@specialstyle : If its value is foo then
7326 %             IF @specialpage = T
7327 %             THEN the command \ps@foo is executed to temporarily
7328 %             reset the page style parameters before composing
7329 %             the current page. This command should execute
7330 %             only \def's and \edef's, making only local
7331 %             definitions.
7332 %
7333 %
7334 % FLOAT PLACEMENT PARAMETERS
7335 %
7336 % The following parameters are set by the macro \@floatplacement.
7337 % When \@floatplacement is called,
7338 % \@colht is the height of the page or column being built. I.e.:
7339 %     * For single-column page it equals \textheight.
7340 %     * For double-column page it equals \textheight - height
7341 %     of double-column floats on page.
7342 % Note that some are set globally and some locally:
7343 %     \@topnum :=G Maximum number of floats allowed on the top of a column.
7344 %     \@toproom :=G Maximum amount of top of column devoted to floats--
7345 %     excluding \textfloatsep separation below the floats and
7346 %     \floatsep separation between them. For two-column
7347 %     output, should be computed as a function of \@colht.
7348 %     \@botnum, \@botroom
7349 %     : Analogous to above.
7350 %     \@colnum :=G Maximum number of floats allowed in a column, including
7351 %     in-text floats.
7352 %     \@textmin :=L Minimum amount of text (excluding footnotes) that must
7353 %     appear on a text page. %% 27 Sep 85 : made local to
7354 %     %% \@addtocurcol and \@addtonextcol
7355 %     \@fpmin :=L Minimum height of floats in a float column.
7356 %
7357 % The macro \@dblfloatplacement sets the following parameters.
7358 %     \@dbltopnum :=G Maximum number of double-column floats allowed at the
7359 %     top of a two-column page.
7360 %     \@dbltoproom :=G Maximum height of double-column floats allowed at
7361 %     top of two-column page.
7362 %     \@fpmin :=L Minimum height of floats in a float column.
7363 % It should also perform the following local assignments where necessary
7364 % -- i.e., where the new value differs from the old one:
7365 %     \@fptop :=L \@dblfpptop
7366 %     \@fpsep :=L \@dblfpsep
7367 %     \@fpbot :=L \@dblfpbot
7368 %
7369 % OUTPUT ROUTINE VARIABLES
7370 %
7371 % \@colht : The total height of the current column. In single column
7372 % style, it equals \textheight. In two-column style, it is
7373 % \textheight minus the height of the double-column floats
7374 % on the current page. MUST BE INITIALIZED TO \textheight.
7375 %
7376 % \@colroom : The height available in the current column for text and
7377 % footnotes. It equals \@colht minus the height of all
7378 % floats committed to the top and bottom of the current
7379 % column.
7380 %
7381 % \footins : Footnote insertion number.
7382 %
7383 % \@maxdepth : Saved value of TeX's \maxdepth. Must be set
7384 % when any routine sets \maxdepth.
7385 %
7386 % CALLING THE OUTPUT ROUTINE
7387 % -----
7388 %
7389 % The output routine is called either by TeX's normal page-breaking
7390 % mechanism, or by a macro putting a penalty < or = -10000 in the output
7391 % list. In the latter case, the penalty indicates why the output
7392 % routine was called, using the following code.
7393 %
7394 % penalty reason
7395 % -----
7396 % -10000 \pagebreak
7397 % \newpage
7398 % -10001 \clearpage (called with \penalty -10000 \vbox{} \penalty -10001

```

```

7399 % -10002 float insertion, called from horizontal mode
7400 % -10003 float insertion, called from vertical mode.
7401 % -10004 float insertion.
7402 %
7403 % Note: A float or marginpar puts the following sequence in the output
7404 % list: (i) a penalty of -10004,
7405 % (ii) a null \vbox
7406 % (iii) a penalty of -10002 or -10003.
7407 % This solves two special problems:
7408 % 1. If the float comes right after a \newpage or \clearpage,
7409 % then the first penalty is ignored, but the second one
7410 % invokes the output routine.
7411 % 2. If there is a split footnote on the page, the second 'page'
7412 % puts out the rest of the footnote.
7413 %
7414 % THE OUTPUT ROUTINE
7415 % -----
7416 %
7417 % FUNCTIONS USED IN THE OUTPUT ROUTINE:
7418 %
7419 % \@outputpage : Produces an output page with the contents of box
7420 % \@outputbox as the text part. Also sets
7421 % \@colht :=G \textheight. The page style is determined
7422 % as follows.
7423 % IF @thispagestyle = true
7424 % THEN use \thispagestyle style
7425 % ELSE use ordinary page style.
7426 %
7427 % \@tryfcolumn\FLIST : Tries to form a float column composed of floats from
7428 % \FLIST with with the following parameters:
7429 % \@colht : height of box
7430 % \@fpm : minimum height of floats in the box
7431 % \@fpsep : interfloat space
7432 % \@fptop : glue at top of box
7433 % \@fpbot : glue at bottom of box.
7434 % If it succeeds, then it does the following:
7435 % * \@outputbox :=L the composed float box.
7436 % * @fcolmade :=L true
7437 % * \FLIST :=G \FLIST - floats put in box
7438 % * \@freelist :=G \@freelist + floats put in box
7439 % If it fails, then:
7440 % * @fcolmade :=L false
7441 % NOTE: BIT MUST BE A SINGLE TOKEN!
7442 %
7443 % \@makefcolumn \FLIST : Same as \@tryfcolumn except that it
7444 % fails to make a float column only if \FLIST is empty.
7445 % Otherwise, it makes a float column containing at least
7446 % the first box in \FLIST, disregarding \@fpm.
7447 %
7448 % \@startcolumn :
7449 % Calls \@tryfcolumn\@deferlist{8}. If \@tryfcolumn returns with
7450 % @fcolmade = false, then:
7451 % * Globally sets \@toplist and \@botlist to floats
7452 % from \@deferlist to go at top and bottom of column,
7453 % deleting them from \@deferlist. It does
7454 % this using \@colht as the total height, the page
7455 % style parameters \@floatsep and \@textfloatsep, and
7456 % the float placement parameters \@topnum, \@toproom,
7457 % \@botnum, \@botroom, \@colnum and \textfraction.
7458 % * Globally sets \@colroom to \@colht minus the height
7459 % of the added floats.
7460 %
7461 % \@startdblcolumn :
7462 % Calls \@tryfcolumn\@dbldeferlist{8}. If \@tryfcolumn returns
7463 % with @fcolmade = false, then:
7464 % * Globally sets \@dbltoplist to floats from \@dbldeferlist
7465 % to go at top and bottom of column, deleting them from
7466 % \@dbldeferlist. It does this using \textheight as the
7467 % total height, and the parameters \@dblfloatsep, etc.
7468 % * Globally sets \@colht to \textheight minus the height
7469 % of the added floats.
7470 %
7471 % \@combinefloats : Combines the text from box
7472 % \@outputbox with the floats from \@toplist and \@botlist,

```



```

7473 %           putting the new box in \@outputbox. It uses \floatsep and
7474 %           \textfloatsep for the appropriate separations. It puts the
7475 %           elements of \TOPLIST and \BOTLIST onto \@freelist, and makes
7476 %           those lists null.
7477 %
7478 % \@makecol : Makes the contents of \box255 plus the accumulated
7479 %           footnotes, plus the floats in \@toplist and \@botlist,
7480 %           into a single column of height \@colht, which it puts
7481 %           into box \@outputbox. It puts boxes in \@midlist back
7482 %           onto \@freelist and restores \maxdepth.
7483 %
7484 % \@opcol : Outputs a column whose text is in box \@outputbox
7485 %           If @twocolumn = false, then it calls \@outputpage,
7486 %           sets \@colht :=G \textheight, and calls \@floatplacement.
7487 %
7488 %           If @twocolumn = true, then:
7489 %           If @firstcolumn = true, then it puts box \@outputbox
7490 %           into \@leftcolumn and sets @firstcolumn :=G false.
7491 %
7492 %           If @firstcolumn = false, then it puts out the current
7493 %           two-column page, any possible two-column float pages,
7494 %           and determines \@dbltoplist for the next page.
7495 %
7496 % \@opcol ==
7497 % BEGIN
7498 %   \@mparbottom :=G Opt
7499 %   if @twocolumn = true
7500 %     then %% \@outputdblcol ==
7501 %       if @firstcolumn = true
7502 %         then @firstcolumn :=G false
7503 %           \@leftcolumn :=G \@outputbox
7504 %       else @firstcolumn :=G true
7505 %         \@outputbox := \vbox{
7506 %           \hbox to \textwidth{
7507 %             \hbox to \columnwidth{\box\@leftcolumn
7508 %               \hss}
7509 %             \hfil \vrule width \columnseprule \hfil
7510 %             \hbox to \columnwidth{\box\@outputbox}
7511 %             \hss}
7512 %           }
7513 %         \@combinedblfloats
7514 %         \@outputpage
7515 %         \begingroup
7516 %           \@dblfloatplacement
7517 %           \@startdblcolumn
7518 %           while @fcolmade = true
7519 %             do \@outputpage
7520 %               \@startdblcolumn od
7521 %         \endgroup
7522 %       fi
7523 %     else
7524 %       \@outputpage
7525 %       \@colht :=G \textheight
7526 %     fi
7527 % END
7528 % \@makecol ==
7529 % BEGIN
7530 %   ifvoid \insert\footins
7531 %     then \@outputbox := \box255
7532 %     else \@outputbox := \vbox {\box\maxdepth :=L \maxdepth
7533 %       %added 21 Jan 87
7534 %       \unvbox255
7535 %       \vskip \skip\footins
7536 %       \footnoterule
7537 %       \unvbox\footins
7538 %     }
7539 %   fi
7540 %   \@freelist :=G \@freelist * \@midlist
7541 %   \@midlist :=G empty
7542 %   \@combinefloats
7543 %   \@outputbox := \vbox to \@colht{\box\maxdepth := \maxdepth
7544 %     \@texttop
7545 %     temp :=L \dp\@outputbox
7546 %     \unvbox\@outputbox

```

```

7547 %                               \vskip -temp
7548 %                               \@textbottom}
7549 %   \maxdepth :=G \@maxdepth
7550 %   END
7551 %
7552 % \@outputpage ==
7553 % BEGIN
7554 %   \begingroup           %% added 11 Jun 85 to keep special page
7555 %                       %% declarations local to this output page
7556 %   \catcode'\ := 10     %%make sure space is really a space
7557 %   \- := \@dischyph    %% Added 4 Aug 88 in event output routine
7558 %   \' := \@acci        %% called inside a tabbing environment.
7559 %   \' := \@accii
7560 %   \= := \@acciii
7561 %   if @specialpage = T
7562 %     then @specialpage :=G F
7563 %     execute \ps@[eval(\@specialstyle)] fi
7564 %   if \@twoside = T
7565 %     then if \count0 odd
7566 %         \@thehead      ==L \@oddhead
7567 %         \@thefoot      ==L \@oddfoot
7568 %         \@themargin    ==L \@oddsidemargin
7569 %     else \@thehead     ==L \@evenhead
7570 %         \@thefoot      ==L \@evenfoot
7571 %         \@themargin    ==L \@evensidemargin fi fi
7572 %   \shipout\ vbox
7573 %   {\normalsize         % set fonts size for head and foot
7574 %    \baselineskip :=L \lineskip :=L Opt
7575 %    \par :=L \@@par     %% added 15 Sep 87 for robustness
7576 %    \vskip \topmargin
7577 %    \moveright\@themargin\ vbox
7578 %      { \box\@tempboxa := \vbox to \headheight{\vfil
7579 %        \hbox to \textwidth
7580 %          {\index == \label ==
7581 %            \glossary == \@gobble
7582 %              %% Added 22 Feb 87 as bug fix
7583 %              %% RmS 91/06/21 \glossary added
7584 %              \@thehead}}
7585 %          \dp\@tempboxa := Opt % Don't skip space for descenders in
7586 %          \box\@tempboxa     % running head.
7587 %          \vskip \headsep
7588 %          \box\@outputbox
7589 %          \baselineskip\footskip
7590 %          \hbox to \textwidth{\index == \label == \glossary == \@gobble
7591 %            %% added 22 Feb 87 as bug fix
7592 %            %% RmS 91/06/21 \glossary added
7593 %            \@thefoot}
7594 %        }
7595 %    }
7596 %   \@colht :=G \textheight
7597 %   \endgroup           %% added 11 Jun 85
7598 %   \stepcounter{page}
7599 %   \firstmark ==L \botmark %% So marks work properly on float
7600 %                       %% pages. (14 Jun 85)
7601 %   END
7602 %
7603 % \@startcolumn ==
7604 % BEGIN
7605 %   \@colroom :=G \@colht
7606 %   if \@deferlist is empty
7607 %     then @fcolmade := false
7608 %     else \@tryfcolumn\@deferlist %% else clause == \@xstartcol
7609 %     if @fcolmade = false
7610 %       then \begingroup
7611 %         \@tempb :=L \@deferlist
7612 %         \@deferlist :=G empty
7613 %         \@elt \BOX == BEGIN \@currbox == \BOX % use \gdef
7614 %           \@addtonextcol
7615 %         END == \@scolelt
7616 %       \@tempb
7617 %     \endgroup
7618 %   fi fi
7619 %   END
7620 %

```

```

7621 % \@startdblcolumn ==
7622 % BEGIN
7623 %   \@colht :=G \textheight
7624 %   \@tryfcolumn\@dbldeferlist %% else clause == \@xstartcol
7625 %   if @fcolmade = false
7626 %     then \begingroup
7627 %       \@tempb :=L \@dbldeferlist
7628 %       \@dbldeferlist :=G empty
7629 %       \@elt \BOX == BEGIN \@currbox == \BOX % use \gdef
7630 %                               \@addtodblcol
7631 %                               END == \@sdblcolelt
7632 %
7633 %       \@tempb
7634 %     \endgroup
7635 %   fi fi
7636 % END
7637 % \output ==
7638 % BEGIN
7639 % case of \outputpenalty
7640 %   > -10001 -> \@makecol
7641 %     \@opcol
7642 %     \@floatplacement
7643 %     \@startcolumn
7644 %     while @fcolmade = true
7645 %     do \@opcol
7646 %     \@startcolumn
7647 %     od
7648 %
7649 %   %% \@specialoutput ==
7650 %
7651 %   -10001 -> %% \@docclearpage ==
7652 %     if there are no footnote insertions
7653 %     then unbox the \writes at the head of \box255
7654 %     and throw away the rest
7655 %     \@deferlist :=G \@toplist * \@botlist
7656 %     * \@deferlist
7657 %     \@toplist :=G \@botlist :=G empty
7658 %     \@colroom :=G \@colht
7659 %     if \@currlist not empty
7660 %     then LaTeX error: float(s) lost
7661 %     \@currlist :=G empty
7662 %
7663 %     fi
7664 %     \@makefcolumn\@deferlist
7665 %     while @fcolmade = true
7666 %     do \@opcol
7667 %     \@makefcolumn\@deferlist
7668 %     od
7669 %     if @twocolumn
7670 %     then
7671 %       if @firstcolumn = true
7672 %       then \@dbldeferlist :=G \@dbltoplist *
7673 %         \@dbldeferlist
7674 %       \@dbltoplist :=G empty
7675 %       \@colht :=G \textheight
7676 %       \begingroup
7677 %         \@dblfloatplacement
7678 %         \@makefcolumn\@dbldeferlist
7679 %         while @fcolmade = true
7680 %         do \@outputpage
7681 %         \@makefcolumn\@dbldeferlist
7682 %         od
7683 %       \endgroup
7684 %     else \vbox{\clearpage}
7685 %     fi fi
7686 %   else \box255 := \vbox{\box255\vfil}
7687 %   \@makecol
7688 %   \@opcol
7689 %   \clearpage
7690 % fi
7691 % < -10001 ->
7692 %   if \outputpenalty < -10003
7693 %   then if \outputpenalty < -20000 %% true only at end
7694 %   then \deadcycles := 0
7695 %   fi

```

```

7695 %           box \@holdpg :=G box255
7696 %     else throw away box 255
7697 %       \@pagedp :=L natural depth of box \@holdpg
7698 %       \@pageht :=L natural ht of box \@holdpg
7699 %       \unvbox box \@holdpg %% put text back
7700 %     if \@currlist nonempty
7701 %       then \@currbox :=L head of \@currlist
7702 %         \@currlist :=G tail of \@currlist
7703 %         if \count\@currbox > 0
7704 %           %% Changed 28 Feb 88 so \@pageht and \@pagedp
7705 %           %% aren't changed for a marginal note
7706 %           then %% this is a float
7707 %             if there are footnote insertions
7708 %               then advance \@pageht and \@pagedp and
7709 %               reinsert footnotes
7710 %             fi
7711 %             \@addtocurcol
7712 %           else %% this is a marginal note
7713 %             if there are footnote insertions
7714 %               reinsert footnotes
7715 %             fi
7716 %             \@addmarginpar
7717 %           fi
7718 %         else THIS SHOULDN'T HAPPEN
7719 %       fi
7720 %     if \outputpenalty < 0           %% TO PERMIT PAGE BREAK
7721 %       then \penalty\interlinepenalty fi %% IF \@addtocurcol
7722 %           %% DIDN'T INSERT A PENALTY
7723 %     fi
7724 %   end case
7725 % \vsize :=G if \outputpenalty > -10004 then \@colroom %%normal case
7726 %           else \maxdimen %%processing float
7727 %   fi
7728 % END
7729 %
7730 % \@combinefloats ==
7731 % BEGIN
7732 %   if \@toplist nonempty
7733 %     then %%\@cfla ==
7734 %       \@elt\BOX == BEGIN \@tempbox := \vbox{\unvbox\@tempbox
7735 %                                     \box\BOX
7736 %                                     \vskip \floatsep}
7737 %       END == \@comflelt
7738 %       \@tempbox := null
7739 %       \@toplist
7740 %       \@outputbox := \vbox{\box\maxdepth :=L \maxdepth
7741 %                             %added 21 Jan 87
7742 %                             \unvbox\@tempbox
7743 %                             \vskip - \floatsep
7744 %                             \topfigrule
7745 %                             \vskip \textfloatsep
7746 %                             \unvbox\@outputbox }
7747 %       \@elt == \relax
7748 %       \@freelist :=G \@freelist * \@toplist
7749 %       \@toplist :=G null
7750 %     fi
7751 %   if \@botlist nonempty
7752 %     then %%\@cflb ==
7753 %       \@elt == \@comflelt
7754 %       \@tempbox := null
7755 %       \@botlist
7756 %       \@outputbox := \vbox{\unvbox\@outputbox
7757 %                             \vskip \textfloatsep
7758 %                             \botfigrule
7759 %                             \unvbox\@tempbox
7760 %                             \vskip - \floatsep }
7761 %       \@elt == \relax
7762 %       \@freelist :=G \@freelist * \@botlist
7763 %       \@botlist :=G null
7764 %     fi
7765 %   END
7766 %
7767 % \@combinedblfloats ==
7768 % BEGIN

```

```

7769 %   if \@dbltoplist nonempty
7770 %     then \@elt == \@comdblfloat
7771 %       \@tempbox := null
7772 %       \@dbltoplist
7773 %       \@outputbox := \vbox to \textheight
7774 %         {\boxmaxdepth :=L \maxdepth
7775 %          \unvbox\@tempbox
7776 %          \vskip - \dblfloatsep
7777 %          \dblfigrule
7778 %          \vskip \dbltextfloatsep
7779 %          \box\@outputbox }
7780 %   \@elt == \relax
7781 %   \@freelist :=G \@freelist * \@dbltoplist
7782 %   \@dbltoplist :=G null
7783 % fi
7784 % END
7785 %
7786 %
7787 %   USER COMMANDS THAT CALL OR AFFECT THE OUTPUT ROUTINE
7788 %   -----
7789 %
7790 % \newpage == BEGIN \par\vfil\penalty -10000 END
7791 %
7792 % \clearpage == BEGIN \newpage
7793 %   \write -1{} % Part of hack to make sure no
7794 %   \vbox{} % \write's get lost.
7795 %   \penalty -10001
7796 %   END
7797 %
7798 % \cleardoublepage == BEGIN \clearpage
7799 %   if @twoside = true and c@page is even
7800 %   then \hbox{} \newpage fi
7801 %   END
7802 %
7803 % \twocolumn ==
7804 % BEGIN
7805 %   \clearpage
7806 %   \columnwidth :=G .5(\textwidth - \columnsep)
7807 %   \hsize :=G \columnwidth
7808 %   @twocolumn :=G true
7809 %   @firstcolumn :=G true
7810 %   \@dblfloatplacement
7811 % END
7812 %
7813 % \onecolumn ==
7814 % BEGIN
7815 %   \clearpage
7816 %   \columnwidth :=G \textwidth
7817 %   \hsize :=G \columnwidth
7818 %   @twocolumn :=G false
7819 %   \@floatplacement
7820 % END
7821 %
7822 %
7823 % \topnewpage{BOX} : starts a new page and puts BOX in a parbox of width
7824 % \textwidth across the top. Useful for full-width titles for
7825 % double-column pages.
7826 % SURPRISE: The stretch from \@dbltextfloatsep will be inserted
7827 % between the BOX and the top of the two columns.
7828 %
7829 % \topnewpage{BOX} ==
7830 % BEGIN
7831 %   \clearpage
7832 %   Take \@currbox from \@freelist
7833 %   \box\@currbox :=G \parbox{BOX} \par
7834 %   \vskip - \@dbltextfloatsep}
7835 %   \count\@currbox :=G 2
7836 %   \@dbltopnum :=G 1
7837 %   \@dbltoproom :=G maxdimension
7838 %   \@addtodblcol
7839 %   \vsize :=G \@colht
7840 %   \@colroom :=G \@colht
7841 % END
7842 %

```

```

7843
7844 %           FLOAT-HANDLING MECHANISMS
7845 %           -----
7846 %
7847 % The float environment obtains an insertion number B from the
7848 % \@freelist (see below for a description of list manipulation), puts
7849 % the float into box B and sets \count B to a FLOAT SPECIFIER. For
7850 % a normal (not double-column) float, it then causes a page break
7851 % in one of the following two ways:
7852 % - In outer hmode: \vadjust{\penalty -10002}
7853 % - In vmode :      \penalty -10003.
7854 % For a double-column float, it puts B onto the \@dbldeferlist.
7855 % The float specifier has two components:
7856 % * A PLACEMENT SPECIFICATION, describing where the float may
7857 %   be placed.
7858 % * A TYPE, which is a power of two--e.g., figures might be
7859 %   type 1 floats, tables type 2 floats, programs type 4 floats, etc.
7860 % The float specifier is encoded as follows, where bit 0 is the least
7861 % significant bit.
7862 %
7863 % Bit   Meaning
7864 % ---   -----
7865 % 0     1 iff the float may go where it appears in the text.
7866 % 1     1 iff the float may go on the top of a page.
7867 % 2     1 iff the float may go on the bottom of a page.
7868 % 3     1 iff the float may go on a float page.
7869 % 4     always 1
7870 % 5     1 iff a type 1 float
7871 % 6     1 iff a type 2 float
7872 % etc.
7873 %
7874 % A negative float specifier is used to indicate a marginal note.
7875 %
7876 %           MACROS AND DATA STRUCTURES FOR PROCESSING FLOATS
7877 %           -----
7878 %
7879 % A FLDAT LIST consisting of the floats in boxes \boxa ... \boxN has the form:
7880 %   \@elt \boxa ... \@elt \boxN
7881 % where \boxI is defined by
7882 %   \newinsert\boxI
7883 % Normally, \@elt is \let to \relax. A test can be performed on the entire
7884 % float list by locally \def'ing \@elt appropriately and executing
7885 % the list. This is a lot more efficient than looping through the list.
7886 %
7887 % The following macros are used for manipulating float lists.
7888 %
7889 % \@next \CS \LIST {NONEMPTY}{EMPTY} == %% NOTE: ASSUME \@elt = \relax
7890 % BEGIN assume that \LIST == \@elt \B1 ... \@elt \Bn
7891 %   if n = 0
7892 %     then EMPTY
7893 %     else \CS :=L \B1
7894 %           \LIST :=G \@elt \B2 ... \@elt \Bn
7895 %           NONEMPTY
7896 %   fi
7897 % END
7898 %
7899 %
7900 % \@bitor\NUM\LIST : Globally sets switch @test to the disjunction for all I
7901 %   of bit log2 \NUM of the float specifiers of all the floats in
7902 %   \LIST. I.e., @test is set to true iff there is at least one
7903 %   float in \LIST having bit log2 \NUM of its float specifier
7904 %   equal to 1.
7905 %
7906 % Note: log2 [(\count I)/32] is the bit number corresponding to the
7907 % type of float I. To see if there is any float in \LIST having
7908 % the same type as float I, you run \@bitor with \NUM = [(\count I)/32] * 32.
7909 %
7910 % \@bitor\NUM\LIST ==
7911 % BEGIN
7912 %   @test :=G false
7913 %   { \@elt \CTR == if \count\CTR / \NUM is odd
7914 %     then @test := true   fi
7915 %   \LIST
7916 %   }

```

```

7917 % END
7918 %
7919 %
7920 % \@cons\LIST\NUM : Globally sets \LIST := \LIST * \@elt \NUM
7921 %
7922 % \@cons\LIST\NUM ==
7923 % BEGIN { \@elt == \relax
7924 %     \LIST :=G \LIST \@elt \NUM
7925 % }
7926 %
7927 % BOX LISTS FOR FLOAT-PLACEMENT ALGORITHMS
7928 %
7929 % \@freelist : List of empty boxes for placing new floats.
7930 % \@toplist : List of floats to go at top of current column.
7931 % \@midlist : List of floats in middle of current column.
7932 % \@botlist : List of floats to go at bottom of current column.
7933 % \@deferlist : List of floats to go after current column.
7934 % \@dbltoplist : List of double-col. floats to go at top of current page.
7935 % \@dbldeferlist : List of double-column floats to go on subsequent pages.
7936 %
7937 % FLOAT-PLACEMENT ALGORITHMS
7938 %
7939 % \@tryfcolumn \FLIST ==
7940 % BEGIN
7941 % @fcolmade :=G false
7942 % \@trylist :=G \FLIST
7943 % \@failedlist :=G empty
7944 % \begingroup
7945 % \@elt == \@xtryfc
7946 % \@trylist
7947 % \endgroup
7948 % if @fcolmade = true
7949 % then \@vtryfc \FLIST
7950 % fi
7951 % END
7952 %
7953 % \@vtryfc ==
7954 % BEGIN
7955 % \@outputbox :=G \vbox{
7956 %     \@elt\BOX == BEGIN
7957 %         \@outputbox :=L \vbox{ \unvbox \@outputbox
7958 %             \vskip \@fpsep
7959 %             \box\BOX }
7960 %     END == \@wtryfc
7961 %     \@flsucceed
7962 %     \@outputbox :=G \vbox to \@colht{ \vskip \@fpsep
7963 %         \vskip -\@fpsep
7964 %         \unvbox \@outputbox
7965 %         \vskip \@fpbot }
7966 %     \@elt == \relax
7967 %     \@freelist :=G \@freelist * \@flsucceed
7968 %     \FLIST :=G \@failedlist * \@flfail
7969 % END
7970 %
7971 % \@xtryfc \BOX ==
7972 % BEGIN
7973 % remove first element from \@trylist
7974 % \@currtype := (\count\BOX / 32) * 32
7975 % \@bitor \@currtype \@failedlist % @test := true if type on list
7976 % \@testfp \BOX % @test := true if no p-option
7977 % if ht of \BOX > \@colht
7978 % then @test :=G true
7979 % fi
7980 % if @test = true
7981 % then add \BOX to \@failedlist
7982 % else \@ytryfc \BOX
7983 % fi
7984 % END
7985 %
7986 % \@ytryfc ==
7987 % BEGIN
7988 % \begingroup
7989 % \@flsucceed :=G \@elt\BOX
7990 % \@flfail :=G empty

```

```

7991 % \@tempdima := \ht\BOX
7992 % \@elt == \@ztryfc
7993 % \@trylist
7994 % if \@tempdima > \@fpmin
7995 % then @fcolmade :=G true
7996 % else add \BOX to \@failedlist
7997 % fi
7998 % \endgroup
7999 % if @fcolmade = true then \@elt == \@gobble fi
8000 % END
8001 %
8002 % \@ztryfc \BOX ==
8003 % BEGIN
8004 % \@tempcnta := (\count\BOX / 32) * 32
8005 % \@bitor \@tempcnta {\@failedlist \@flfail} % @test := true if on a list
8006 % \@testfp \BOX % @test := true if not p-option
8007 % \@tempdimb := \@tempdima + ht of \BOX + \@fpsep
8008 % if \@tempdimb > \@colht
8009 % then @test :=G true
8010 % fi
8011 % if @test = true
8012 % then add \BOX to \@flfail
8013 % else add \BOX to \@flsucceed
8014 % \@tempdima := \@tempdimb
8015 % fi
8016 % END
8017 %
8018 % \@testfp \BOX == BEGIN if bit 3 of \count\BOX = 0
8019 % then @test :=G true fi
8020 % END
8021 %
8022 % \@makefcolumn \FLIST ==
8023 % BEGIN
8024 % \beginngroup
8025 % \@fpmin =:L 0
8026 % \@testfp == \@gobble
8027 % \@tryfcolumn \FLIST
8028 % \endngroup
8029 % END
8030 %
8031 % \@addtobot : Tries to put insert \@currbox on \@botlist. Called only when:
8032 % * \ht \BOX + \@maxsep < \@colroom
8033 % * type of \@currbox not on \@deferlist
8034 % * \@colnum > 0
8035 % * @insert = false
8036 % If it succeeds, then:
8037 % * sets @insert true
8038 % * decrements \@botroom by \ht \BOX
8039 % * decrements \@botnum and \@colnum by 1
8040 % * decrements \@colroom by \ht \BOX + either \floatsep
8041 % or \textfloatsep, as appropriate.
8042 % * sets \maxdepth to Opt
8043 %
8044 % \@addtotoporbot : Tries to put insert \@currbox on \@toplist or \@botlist.
8045 % Called only under same conditions as \@addtobot.
8046 % If it succeeds, then:
8047 % * sets @insert true
8048 % * decrements either \@toproom or \@botroom by \ht \BOX
8049 % * decrements \@colnum and either \@topnum or
8050 % \@botnum by 1
8051 % * decrements \@colroom by \ht \BOX + either \floatsep
8052 % or \textfloatsep, as appropriate.
8053 %
8054 % \@addtocurcol : Tries to add \@currbox to current column, setting @insert
8055 % true if it succeeds, false otherwise. It will add
8056 % \@currbox to top only if bit 0 of \count \@currbox is 0, and
8057 % to the bottom only if bit 0 = 0 or an earlier float of
8058 % the same type is put on the bottom.
8059 % If the float is put in the text, then
8060 % \penalty\interlinepenalty is put
8061 % right after the float, before the following \uskip, and
8062 % \outputpenalty :=L 0.
8063 %
8064 % \@addtonextcol : Tries to add \@currbox to the next column, setting @insert

```



```

8065 %           true if it succeeds, false otherwise.
8066 %
8067 % \@addtodblcol : Tries to add \@currbox to the next double-column page,
8068 % adding it to \@dbltoplist if it succeeds and \@dbldeferlist
8069 % if it fails.
8070 %
8071 % \@addtobot ==
8072 % BEGIN
8073 %   if bit 2 of \count \@currbox = 1
8074 %   then if \@botnum > 0
8075 %       then if \@botroom > \ht \@currbox
8076 %           then \@botnum :=G \@botnum - 1
8077 %               \@colnum :=G \@colnum - 1
8078 %               \@tempdima :=L - \ht \@currbox -
8079 %                   if \@botlist empty
8080 %                       then \textfloatsep
8081 %                   else \floatsep
8082 %                   fi
8083 %               \@botroom :=G \@botroom + \@tempdima
8084 %               \@colroom :=G \@colroom + \@tempdima
8085 %               add \@currbox to \@botlist
8086 %               \maxdepth :=G Opt
8087 %               @insert :=L true
8088 %           fi
8089 %       fi
8090 %   END
8091 % \@addtotoporbot ==
8092 % BEGIN
8093 %   if bit 1 of \count \@currbox = 1
8094 %   then if \@topnum > 0
8095 %       then if \@toproom > \ht \@currbox
8096 %           then if \@currtype not on \@midlist or \@botlist
8097 %               then \@topnum :=G \@topnum - 1
8098 %                   \@colnum :=G \@colnum - 1
8099 %                   \@tempdima :=L - \ht \@currbox -
8100 %                       if \@toplist empty
8101 %                           then \textfloatsep
8102 %                       else \floatsep
8103 %                       fi
8104 %                   \@toproom :=G \@toproom + \@tempdima
8105 %                   \@colroom :=G \@colroom + \@tempdima
8106 %                   add \@currbox to \@toplist
8107 %                   @insert :=L true
8108 %           fi
8109 %       fi
8110 %   END
8111 %
8112 % \@addtocurcol ==
8113 % BEGIN
8114 %   @insert :=L false
8115 %   \@textmin := \textfraction\@colht %% added 27 Sep 85
8116 %   if \@colroom > \ht \@currbox + max(\@pageht+\@pagedp, \@textmin)
8117 %       + \@maxsep
8118 %   then if \@colnum > 0
8119 %       then \@currtype := type of \@currbox
8120 %           if \@currtype not on \@deferlist
8121 %               then if \@currtype on \@botlist
8122 %                   then \@addtobot
8123 %                   else if bit 0 of \count \@currbox = 1
8124 %                       then decrement \@colnum
8125 %                           put \@currbox on \@midlist
8126 %                           add \@currbox + space +
8127 %                               \penalty \interlinepenalty to text
8128 %                               \outputpenalty :=L 0
8129 %                           @insert := true
8130 %                       else \@addtotoporbot
8131 %                   fi
8132 %               fi
8133 %           then add \@currbox to \@deferlist
8134 %           fi
8135 %   END
8136 %
8137 % \@addtonextcol ==
8138 % BEGIN

```

```

8139 % @insert :=L false
8140 % \textmin := \textfraction\colht %% added 27 Sep 85
8141 % if \colroom > \ht \currbox + \textmin + \maxsep
8142 %   then if \colnum > 0
8143 %     \currtype := type of \currbox
8144 %     then if \currtype not on \deferlist
8145 %       then \addtotoporbot
8146 %   fi fi fi
8147 % if @insert = false
8148 %   then add \currbox to \deferlist
8149 % fi
8150 % END
8151 %
8152 % \addtodblcol ==
8153 % BEGIN
8154 % @insert :=L false
8155 % if bit 1 of \count \currbox = 1
8156 %   then if \dbltopnum > 0
8157 %     then if \dbltoproom > \ht \currbox
8158 %       then if type of \currbox not on \dbldeferlist
8159 %         then \dbltopnum :=G \dbltopnum - 1
8160 %           \tempdima := -\ht\currbox -
8161 %             if \dbltoplist empty
8162 %               then \dbltextfloatsep
8163 %             else \dblfloatsep
8164 %           fi
8165 %           \dbltoproom :=G \dbltoproom+\tempdima
8166 %           \colht :=G \colht+\tempdima
8167 %           add \currbox to \dbltoplist
8168 %           @insert :=L true
8169 %         fi fi fi fi
8170 %       if @insert = false then add \currbox to \dbldeferlist
8171 %     END
8172 %
8173 % \addmarginpar ==
8174 % BEGIN
8175 % if \currlist nonempty
8176 %   then remove \marbox from \currlist %% NOTE: \currbox = left box
8177 %   add \marbox and \currbox to \freelist
8178 %   else LaTeX error: ? %% shouldn't happen
8179 % fi
8180 % \tempcnta := 1 %% 1 = right, -1 = left
8181 % if @twocolumn = true
8182 %   then if @firstcolumn = true
8183 %     then \tempcnta := -1
8184 %   fi
8185 %   else if @mparswitch = true
8186 %     then if count0 odd
8187 %       else \tempcnta := -1
8188 %     fi
8189 %   fi
8190 %   if @reversemargin = true
8191 %     then \tempcnta := -\tempcnta
8192 %   fi
8193 % fi
8194 % if \tempcnta < 0 then \box\marbox :=G \box\currbox fi
8195 % \tempdima :=L maximum(\mparbottom - \pageht + ht of \marbox, 0)
8196 % if \tempdima > 0 then LaTeX warning: 'marginpar moved' fi
8197 % \mparbottom :=G \pageht + \tempdima + depth of \marbox
8198 %   + \marginparpush
8199 % \tempdima :=L \tempdima - ht of \marbox
8200 % height of \marbox :=G depth of \marbox :=G 0
8201 % \skip -\pagedp
8202 % \skip \tempdima
8203 % \nointerlineskip
8204 % \hbox{ if \tempcnta > 0 then \hskip \columnwidth
8205 %   \hskip \marginparsep
8206 %   else \hskip -\marginparsep
8207 %   \hskip -\marginparwidth
8208 % fi
8209 % \box\marbox
8210 % \hss
8211 % }
8212 % \skip -\tempdima

```

```

8213 % \nointerlineskip
8214 % \hbox{\vrule height 0 width 0 depth \@pagedp}
8215 % END
8216
8217
8218 \maxdeadcycles = 100 % floats and \marginpar's add a lot of dead cycles
8219
8220 \let\@elt\relax
8221
8222 \def\@next#1#2#3#4{\ifx#2\@empty #4\else
8223   \expandafter\@xnext #2\@#1#2#3\fi}
8224
8225 \def\@xnext \@elt #1#2\@#3#4{\def#3{#1}\gdef#4{#2}}
8226
8227 \newif\if@test
8228
8229 \def\@bitor#1#2{\global\@testfalse {\let\@elt\@xbitor
8230   \@tempcnta #1\relax #2}}
8231
8232 % RmS 91/11/22: Added test for \count#1 being 0.
8233 %% Suggested by Chris Rowley.
8234 \def\@xbitor #1{\@tempcntb \count#1
8235   \ifnum \@tempcnta =\z@
8236     \else
8237       \divide\@tempcntb\@tempcnta
8238       \ifodd\@tempcntb \global\@testtrue\fi
8239     \fi}
8240
8241 % DEFINITION OF FLOAT BOXES:
8242 \newinsert\bx@A
8243 \newinsert\bx@B
8244 \newinsert\bx@C
8245 \newinsert\bx@D
8246 \newinsert\bx@E
8247 \newinsert\bx@F
8248 \newinsert\bx@G
8249 \newinsert\bx@H
8250 \newinsert\bx@I
8251 \newinsert\bx@J
8252 \newinsert\bx@K
8253 \newinsert\bx@L
8254 \newinsert\bx@M
8255 \newinsert\bx@N
8256 \newinsert\bx@O
8257 \newinsert\bx@P
8258 \newinsert\bx@Q
8259 \newinsert\bx@R
8260
8261
8262
8263 \gdef\@freelist{\@elt\bx@A\@elt\bx@B\@elt\bx@C\@elt\bx@D\@elt\bx@E
8264   \@elt\bx@F\@elt\bx@G\@elt\bx@H\@elt\bx@I\@elt\bx@J
8265   \@elt\bx@K\@elt\bx@L\@elt\bx@M\@elt\bx@N
8266   \@elt\bx@O\@elt\bx@P\@elt\bx@Q\@elt\bx@R}
8267
8268 \gdef\@toplist{}
8269 \gdef\@botlist{}
8270 \gdef\@midlist{}
8271 \gdef\@currlist{}
8272 \gdef\@deferlist{}
8273 \gdef\@dbltoplist{}
8274 \gdef\@dbldeferlist{}
8275
8276 % PAGE LAYOUT PARAMETERS
8277 \newdimen\topmargin
8278 \newdimen\oddsidemargin
8279 \newdimen\evensidemargin
8280 \let\@themargin=\oddsidemargin
8281 \newdimen\headheight
8282 \newdimen\headsep
8283 \newdimen\footskip
8284 \newdimen\footheight % even though it never gets used.
8285 \newdimen\textheight
8286 \newdimen\textwidth

```

```

8287 \newdimen\columnwidth
8288 \newdimen\columnsep
8289 \newdimen\columnseprule
8290 \newdimen\@maxdepth \@maxdepth = \maxdepth
8291 \newdimen\marginparwidth
8292 \newdimen\marginparsep
8293 \newdimen\marginparpush
8294
8295 % PAGE STYLE PARAMETERS
8296 \newskip\floatsep
8297 \newskip\textfloatsep
8298 \newskip\intextsep
8299 \newdimen\@maxsep
8300 \newskip\dblfloatsep
8301 \newskip\dbltextfloatsep
8302 \newdimen\@dblmaxsep
8303 \newskip\@fptop
8304 \newskip\@fpsep
8305 \newskip\@fpbot
8306 \newskip\@dblftop
8307 \newskip\@dblfpsep
8308 \newskip\@dblfpbot
8309 \let\topfigrule=\relax
8310 \let\botfigrule=\relax
8311 \let\dblfigrule=\relax
8312
8313 % INTERNAL REGISTERS
8314
8315 \newcount\@topnum
8316 \newdimen\@toproom
8317 \newcount\@dbltopnum
8318 \newdimen\@dbltoproom
8319 \newcount\@botnum
8320 \newdimen\@botroom
8321 \newcount\@colnum
8322 \newdimen\@textmin
8323 \newdimen\@fpmin
8324 \newdimen\@colht
8325 \newdimen\@colroom
8326 \newdimen\@pageht
8327 \newdimen\@pagedp
8328 \newdimen\@mparbottom \@mparbottom\z@
8329 \newcount\@currtype
8330 \newbox\@outputbox
8331 \newbox\@leftcolumn
8332 \newbox\@holdpg
8333
8334 \newif\if@insert
8335 \newif\if@fcolmade
8336 \newif\if@specialpage \@specialpagefalse
8337 \newif\if@twoside \@twosidefalse
8338 \newif\if@firstcolumn \@firstcolumntrue
8339 \newif\if@twocolumn \@twocolumnfalse
8340 \newif\if@reversemargin \@reversemarginfalse
8341 \newif\if@mparswitch \@mparswitchfalse
8342
8343 \def\@thehead{\@oddhead} % initialization
8344 \def\@thefoot{\@oddfoot}
8345
8346 \def\newpage{\par\vfil\penalty -\@M}
8347
8348 \def\clearpage{\newpage \write\m@ne{}\vbox{} \penalty -\@Mi}
8349
8350 \def\cleardoublepage{\clearpage\if@twoside \ifodd\c@page\else
8351 \hbox{} \newpage\if@twocolumn\hbox{} \newpage\fi\fi\fi}
8352
8353 \def\twocolumn{\clearpage \global\columnwidth\textwidth
8354 \global\advance\columnwidth -\columnsep \global\divide\columnwidth\tw@
8355 \global\hsize\columnwidth \global\linewidth\columnwidth
8356 \global\@twocolumntrue \global\@firstcolumntrue
8357 \@dblfloatplacement\@ifnextchar[{\@topnewpage}{}}
8358
8359 \def\onecolumn{\clearpage\global\columnwidth\textwidth
8360 \global\hsize\columnwidth \global\linewidth\columnwidth

```

```

8361     \global\@twocolumnfalse \@floatplacement}
8362
8363 \long\def\@topnewpage[#1]{\@next\@currbox\@freelist{}}%
8364     \global\setbox\@currbox\vbox{\hsize\textwidth \@parboxrestore
8365     #1\par\vskip -\dbltextfloatsep}\global\count\@currbox\@tw@
8366     \global\@dbltopnum\@ne \global\@dbltoproom\maxdimen\@addtodblcol
8367     \global\vsize\@colht \global\@colroom\@colht}
8368
8369 % RmS 91/09/29: added reset of \par to the output routine.
8370 %%
8371 %%      This avoids problems when the output routine is
8372 %%      called within a list where \par may be a no-op.
8373
8373 \output{\let\par\@@par
8374     \ifnum\outputpenalty <-\@M\@specialoutput\else
8375     \@makecol\@opcol\@floatplacement\@startcolumn
8376     \@whiles\if@fcolmade \fi{\@opcol\@startcolumn}\fi
8377     \global\vsize\ifnum\outputpenalty >-\@Miv \@colroom
8378     \else \maxdimen\fi}
8379
8380 % CHANGES TO \@specialoutput:
8381 % * \penalty\z@ changed to \penalty\interlinepenalty so \samepage
8382 % works properly with figure and table environments.
8383 % (Changed 23 Oct 86)
8384 %
8385 % * Definition of \@specialoutput changed 26 Feb 88 so \@pageht and \@pagedp
8386 % aren't changed for a marginal note. (Change suggested by
8387 % Chris Rowley.)
8388 %
8389 \def\@specialoutput{\ifnum\outputpenalty >-\@Mii
8390     \@doclearpage
8391     \else
8392     \ifnum \outputpenalty <-\@Miii
8393     \ifnum\outputpenalty<-\@MM \deadcycles\z@\fi
8394     \global\setbox\@holdpg\vbox{\unvbox\@cclv}%
8395     \else \setbox\@tempboxa\box\@cclv
8396     \@pagedp\dp\@holdpg \@pageht\ht\@holdpg
8397     \unvbox\@holdpg
8398     \@next\@currbox\@currlist{\ifnum\count\@currbox >\z@
8399     \ifvoid\footins\else\advance\@pageht\ht\footins
8400     \advance\@pageht\skip\footins \advance\@pagedp\dp\footins
8401     \insert\footins{\unvbox\footins}\fi
8402     \@addtocurcol\else
8403     \ifvoid\footins\else\insert\footins{\unvbox\footins}\fi
8404     \@addmarginpar\fi}\@latexbug
8405     \ifnum \outputpenalty <\z@ \penalty\interlinepenalty\fi
8406     \fi\fi}
8407
8408
8409 \def\@doclearpage{\ifvoid\footins
8410     \setbox\@tempboxa\vsplit\@cclv to\z@ \unvbox\@tempboxa
8411     \setbox\@tempboxa\box\@cclv
8412     \xdef\@deferlist{\@toplist\@botlist
8413     \@deferlist}\gdef\@toplist{}\gdef\@botlist{}\global\@colroom\@colht
8414     \ifx\@currlist
8415     \empty\else\@latexerr{Float(s)
8416     lost}\@ehb\gdef\@currlist{}\fi
8417     \@makefcolumn\@deferlist
8418     \@whiles\if@fcolmade \fi{\@opcol
8419     \@makefcolumn\@deferlist}\if@twocolumn
8420
8421     \if@firstcolumn
8422     \xdef\@dbldeferlist{\@dbltoplist
8423     \@dbldeferlist}\gdef\@dbltoplist{}\global\@colht\textheight
8424     \begingroup \@dblfloatplacement \@makefcolumn\@dbldeferlist
8425     \@whiles\if@fcolmade \fi{\@outputpage
8426     \@makefcolumn\@dbldeferlist}\endgroup
8427     \else \vbox{}\clearpage
8428     \fi\fi
8429     \else\setbox\@cclv\vbox{\box\@cclv\vfil}\@makecol\@opcol
8430     \clearpage
8431     \fi}
8432
8432 \def\@opcol{\global\@mparbottom\z@\if@twocolumn\@outputdblcol\else
8433     \@outputpage \global\@colht\textheight \fi}
8434

```

```

8435 \def\@outputdblcol{\if@firstcolumn \global\@firstcolumnfalse
8436 \global\setbox\@leftcolumn\box\@outputbox
8437 \else \global\@firstcolumntrue
8438 \setbox\@outputbox\vbox{\hbox to\textwidth{\hbox to\columnwidth
8439 {\box\@leftcolumn \hss}\hfil \vrule width\columnseprule\hfil
8440 \hbox to\columnwidth{\box\@outputbox \hss}}}\@combinedblfloats
8441 \@outputpage \begingroup \@dblfloatplacement \@startdblcolumn
8442 \@whiles\if@colmade \fi{\@outputpage\@startdblcolumn}\endgroup
8443 \fi}
8444
8445 % Extra \@texttop somehow found its way into \@makecol. Deleted
8446 % 1 Dec 86. (Found by Mike Harrison)
8447 %% RmS 91/10/22: Replaced \dimen128 by \dimen@.
8448 \def\@makecol{\ifvoid\footins \setbox\@outputbox\box\@cclv
8449 \else\setbox\@outputbox
8450 \vbox{\boxmaxdepth \maxdepth
8451 \unvbox\@cclv\vskip\skip\footins\footnoterule\unvbox\footins}\fi
8452 \xdef\@freelist{\@freelist\@midlist}\gdef\@midlist{}\@combinefloats
8453 \setbox\@outputbox\vbox to\@colht{\boxmaxdepth\maxdepth
8454 \@texttop\dimen@dp\@outputbox\unvbox\@outputbox
8455 \vskip-\dimen@\@textbottom}%
8456 \global\maxdepth\@maxdepth}
8457
8458 \let\@texttop=\relax
8459 \let\@textbottom=\relax
8460
8461 \def\@outputpage{\begingroup\catcode'\ =10
8462 \let-\@dischyp\let'\@acci\let'\@accii\let'\@acciii
8463 \if@specialpage
8464 \global\@specialpagefalse\@nameuse{ps@\@specialstyle}\fi
8465 \if@twoside
8466 \ifodd\count\z@ \let\@thehead\@oddhead \let\@thefoot\@oddfoot
8467 \let\@themargin\oddsidemargin
8468 \else \let\@thehead\@evenhead
8469 \let\@thefoot\@evenfoot \let\@themargin\evensidemargin
8470 \fi\fi
8471 \shipout
8472 \vbox{\reset@font %% RmS 91/08/15
8473 \normalsize \baselineskip\z@ \lineskip\z@
8474 \let\par\@par %% 15 Sep 87
8475 \vskip \topmargin \moveright\@themargin
8476 \vbox{\setbox\@tempboxa
8477 \vbox to\headheight{\vfil \hbox to\textwidth
8478 {\let\label\@gobble \let\index\@gobble
8479 \let\glossary\@gobble %% 21 Jun 91
8480 \@thehead}}}% %% 22 Feb 87
8481 \dp\@tempboxa\z@
8482 \box\@tempboxa
8483 \vskip \headsep
8484 \box\@outputbox
8485 \baselineskip\footskip
8486 \hbox to\textwidth{\let\label\@gobble
8487 \let\index\@gobble %% 22 Feb 87
8488 \let\glossary\@gobble %% 21 Jun 91
8489 \@thefoot}}}\global\@colht\textheight
8490 \endgroup\stepcounter{page}\let\firstmark\botmark}
8491
8492
8493 \def\@combinefloats{\boxmaxdepth\maxdepth \ifx\@toplist\@empty\else\@cfla\fi
8494 \ifx\@botlist\@empty\else\@cflb\fi}
8495
8496 \def\@cfla{\let\@elt\@comflelt \setbox\@tempboxa\vbox{}\@toplist
8497 \setbox\@outputbox\vbox{\unvbox\@tempboxa\vskip-\floatsep
8498 \topfigrule\vskip\textfloatsep \unvbox\@outputbox}\let\@elt\relax
8499 \xdef\@freelist{\@freelist\@toplist}\gdef\@toplist{}}
8500
8501 \def\@cflb{\let\@elt\@comflelt \setbox\@tempboxa\vbox{}\@botlist
8502 \setbox\@outputbox\vbox{\unvbox\@outputbox \vskip\textfloatsep
8503 \botfigrule\unvbox\@tempboxa \vskip-\floatsep}\let\@elt\relax
8504 \xdef\@freelist{\@freelist\@botlist}\gdef\@botlist{}}
8505
8506 \def\@comflelt#1{\setbox\@tempboxa
8507 \vbox{\unvbox\@tempboxa\box #1\vskip\floatsep}}
8508

```

```

8509 \def\@combinedblfloats{\ifx\@dbltoplist\@empty\else
8510   \let\@elt\@comdblfilelt \setbox\@tempboxa\vbox{}\@dbltoplist
8511   \setbox\@outputbox\vbox to\textheight
8512     {\boxmaxdepth\maxdepth
8513     \unvbox\@tempboxa\vskip-\dblfloatsep
8514     \dblfigrule\vskip\dbltextfloatsep \box\@outputbox}\let\@elt\relax
8515   \xdef\@freelist{\@freelist\@dbltoplist}\gdef\@dbltoplist{}\fi}
8516
8517
8518 \def\@comdblfilelt#1{\setbox\@tempboxa
8519   \vbox{\unvbox\@tempboxa\box #1\vskip\dblfloatsep}}
8520
8521
8522 \def\@startcolumn{\global\@colroom\@colht
8523   \ifx\@deferlist\@empty\global\@fcolmadefalse\else\@xstartcol\fi}
8524
8525 \def\@xstartcol{\@tryfcolumn\@deferlist \if@fcolmade\else
8526   \begingroup\edef\@tempb{\@deferlist}\gdef\@deferlist{}\let\@elt\@scolet
8527   \@tempb\endgroup\fi}
8528
8529 \def\@scolet#1{\def\@currbox{#1}\@addtonextcol}
8530
8531 \def\@startdblcolumn{\global\@colht\textheight
8532   \@tryfcolumn\@dbldeferlist \if@fcolmade\else
8533   \begingroup
8534     \edef\@tempb{\@dbldeferlist}\gdef\@dbldeferlist{}\let\@elt\@sdblcolet
8535     \@tempb\endgroup\fi}
8536
8537 \def\@sdblcolet#1{\def\@currbox{#1}\@addtodblcol}
8538
8539 \def\@tryfcolumn #1{\global\@fcolmadefalse \xdef\@trylist{#1}\xdef\@failedlist
8540   }\begingroup \let\@elt\@xtryfc \@trylist \endgroup
8541   \if@fcolmade \@vtryfc #1\fi}
8542
8543 \def\@vtryfc #1{\global\setbox\@outputbox\vbox{}\let\@elt\@wtryfc
8544   \@flsucceed \global\setbox\@outputbox\vbox to\@colht{\vskip \@fptop
8545   \vskip -\@fpsep \unvbox \@outputbox \vskip \@fpbot}\let\@elt\relax
8546   \xdef #1{\@failedlist\@flfail}\xdef\@freelist{\@freelist\@flsucceed}}
8547
8548 \def\@wtryfc #1{\global\setbox\@outputbox\vbox{\unvbox\@outputbox
8549   \vskip\@fpsep\box #1}}
8550
8551
8552 \def\@xtryfc #1{\@next\@tempa\@trylist{}}\@currtype
8553   \count #1\divide\@currtype\@xxxii \multiply\@currtype\@xxxii
8554   \@bitor \@currtype \@failedlist \@testfp #1\ifdim
8555     \ht #1>\@colht \global\@testtrue\fi
8556   \if@test \@cons\@failedlist #1\else \@ytryfc #1\fi}
8557
8558 \def\@ytryfc #1{\begingroup \gdef\@flsucceed{\@elt #1}\gdef\@flfail
8559   }\@tempdima\ht #1\let\@elt\@ztryfc \@trylist \ifdim \@tempdima >\@fpmi
8560   \global\@fcolmadetrue \else \@cons\@failedlist #1\fi
8561   \endgroup \if@fcolmade \let\@elt\@gobble \fi}
8562
8563 \def\@ztryfc #1{\@tempcnta\count #1\divide\@tempcnta\@xxxii
8564   \multiply\@tempcnta\@xxxii \@bitor \@tempcnta {\@failedlist
8565   \@flfail}\@testfp #1\@tempdimb\@tempdima \advance\@tempdimb\ht #1\advance
8566   \@tempdimb\@fpsep \ifdim \@tempdimb >\@colht \global\@testtrue\fi
8567   \if@test \@cons\@flfail #1\else \@cons\@flsucceed #1\@tempdima\@tempdimb
8568   \fi}
8569
8570 \def\@testfp #1{\@tempcnta\count #1\divide\@tempcnta 8\relax
8571   \ifodd\@tempcnta \else \global\@testtrue\fi}
8572
8573 \def\@makefcolumn #1{\begingroup \@fpmi\z@ \let\@testfp\@gobble
8574   \@tryfcolumn #1\endgroup}
8575
8576 \def\@addtobot{\@tempcnta\count\@currbox\divide\@tempcnta4 \ifodd\@tempcnta
8577   \ifnum \@botnum >\z@ \ifdim \@botroom >\ht\@currbox
8578     \global\advance\@botnum\m@ne
8579     \global\advance\@colnum\m@ne
8580     \@tempdima -\ht\@currbox
8581     \advance\@tempdima -\ifx\@botlist\@empty \textfloatsep
8582     \else\floatsep\fi}

```

```

8583 \global\advance\@botroom \@tempdima
8584 \global\advance\@colroom \@tempdima
8585 \@cons\@botlist\@currbox \global\maxdepth\z@
8586 \@inserttrue\fi\fi\fi}
8587
8588 \def\@addtotoporbot{\@tempcnta\count\@currbox \divide\@tempcnta\tw@
8589 \ifodd\@tempcnta \ifnum \@topnum >\z@ \ifdim\@toproom >\ht\@currbox
8590 \@bitor\@currtype{\@midlist\@botlist}\if@test\else
8591 \global\advance\@topnum\m@ne
8592 \global\advance\@colnum\m@ne
8593 \@tempdima-\ht\@currbox
8594 \advance\@tempdima
8595 -\ifx\@toplist\@empty \textfloatsep \else\floatsep\fi
8596 \global\advance\@toproom \@tempdima
8597 \global\advance\@colroom \@tempdima
8598 \@cons\@toplist\@currbox
8599 \@inserttrue
8600 \fi\fi\fi\fi
8601 \if@insert\else\@addtobot \fi}
8602
8603 \def\@addtonextcol{\@insertfalse \@textmin \textfraction\@colht
8604 \@tempdima\ht\@currbox
8605 \advance\@tempdima\@textmin\advance\@tempdima\@maxsep
8606 \ifdim\@colroom >\@tempdima
8607 \ifnum\@colnum >\z@
8608 \@currtype\count\@currbox \divide\@currtype\@xxxii
8609 \multiply\@currtype\@xxxii
8610 \@bitor\@currtype\@deferlist
8611 \if@test\else
8612 \@addtotoporbot
8613 \fi\fi\fi
8614 \if@insert\else \@cons\@deferlist\@currbox\fi}
8615
8616 \def\@addtodblcol{\@insertfalse
8617 \@tempcnta\count\@currbox \divide\@tempcnta\tw@
8618 \ifodd\@tempcnta
8619 \ifnum\@dbltopnum >\z@
8620 \ifdim\@dbltoproom >\ht\@currbox
8621 \@currtype\count\@currbox \divide\@currtype\@xxxii
8622 \multiply\@currtype\@xxxii
8623 \@bitor\@currtype\@dbldeferlist
8624 \if@test\else
8625 \global\advance\@dbltopnum\m@ne
8626 \@tempdima -\ht\@currbox
8627 \advance\@tempdima -\ifx\@dbltoplist\@empty
8628 \dbltextfloatsep\else\dblfloatsep\fi
8629 \global\advance\@dbltoproom \@tempdima
8630 \global\advance\@colht \@tempdima
8631 \@cons\@dbltoplist\@currbox
8632 \@inserttrue
8633 \fi\fi\fi\fi
8634 \if@insert\else \@cons\@dbldeferlist\@currbox \fi}
8635
8636 % CHANGE TO \@addtocurcol:
8637 % \penalty\z@ changed to \penalty\interlinepenalty so \samepage
8638 % works properly with figure and table environments.
8639 % (Changed 23 Oct 86)
8640 %
8641 \def\@addtocurcol{\@insertfalse \@textmin \textfraction\@colht
8642 \@tempdima\@pageht \advance\@tempdima\@pagedp
8643 \ifdim \@textmin >\@tempdima \@tempdima\@textmin \fi
8644 \advance\@tempdima\ht\@currbox \advance\@tempdima\@maxsep
8645 \ifdim\@colroom >\@tempdima
8646 \ifnum\@colnum >\z@
8647 \@currtype\count\@currbox \divide\@currtype\@xxxii
8648 \multiply\@currtype\@xxxii
8649 \@bitor\@currtype\@deferlist
8650 \if@test\else
8651 \@bitor\@currtype\@botlist
8652 \if@test \@addtobot \else
8653 \ifodd\count\@currbox
8654 \global\advance\@colnum\m@ne
8655 \@cons\@midlist\@currbox
8656 \vskip\intextsep \box\@currbox

```



```

8657         \penalty\interlinepenalty \vskip\intextsep
8658         \ifnum\outputpenalty <-\@Mii \vskip -\parskip\fi
8659         \outputpenalty\z@
8660         \@inserttrue
8661         \else \@addtotoporbot
8662     \fi\fi\fi\fi\fi
8663     \if@insert\else\@cons\@deferlist\@currbox\fi}
8664
8665 \def\@addmarginpar{\@next\@marbox\@currlist{\@cons\@freelist\@marbox
8666 \@cons\@freelist\@currbox}\@latexbug\@tempcnta\@ne
8667 \if@twocolumn
8668     \if@firstcolumn \@tempcnta\m@ne \fi
8669 \else
8670     \if@mparswitch
8671         \ifodd\c@page \else\@tempcnta\m@ne \fi
8672     \fi
8673     \if@reversemargin \@tempcnta -\@tempcnta \fi
8674 \fi
8675 \ifnum\@tempcnta <\z@ \global\setbox\@marbox\box\@currbox \fi
8676 \@tempdima\@mparbottom \advance\@tempdima -\@pageht
8677 \advance\@tempdima\ht\@marbox \ifdim\@tempdima >\z@
8678     \@@warning{Marginpar on page \thepage\space moved}\else\@tempdima\z@ \fi
8679 \global\@mparbottom\@pageht \global\advance\@mparbottom\@tempdima
8680 \global\advance\@mparbottom\dp\@marbox
8681 \global\advance\@mparbottom\marginparpush
8682 \advance\@tempdima -\ht\@marbox
8683 \global\ht\@marbox\z@ \global\dp\@marbox\z@
8684 \vskip -\@pagedp \vskip\@tempdima\nointerlineskip
8685 \hbox to\columnwidth
8686     {\ifnum \@tempcnta >\z@
8687         \hskip\columnwidth \hskip\marginparsep
8688         \else \hskip -\marginparsep \hskip -\marginparwidth \fi
8689         \box\@marbox \hss}\nobreak %% RmS 91/06/21 \nobreak added
8690 \vskip -\@tempdima
8691 \nointerlineskip
8692 \hbox{\vrule \@height\z@ \@width\z@ \@depth\@pagedp}}
8693
8694 \message{debugging}
8695 % *****
8696 % * DEBUGGING AND TEST INITIALIZATIONS *
8697 % *****
8698 %
8699 % DEBUGGING
8700 \def\showoverfull{\tracingonline=1}
8701 \tracingstats1 % SHOWS HOW MUCH STUFF TeX HAS USED
8702 \def\showoutput{\tracingonline1\tracingoutput1
8703 \showboxbreadth99999\showboxdepth99999\errorstopmode}
8704 \def\makeatletter{\catcode'\@=11\relax}
8705 \def\makeatother{\catcode'\@=12\relax}
8706
8707 \newcount\@lowpenalty
8708 \newcount\@medpenalty
8709 \newcount\@highpenalty
8710
8711 % LIST
8712
8713 % ENUMERATION
8714
8715 % ITEMIZE
8716
8717 % ARRAY AND TABULAR
8718
8719 % THE PICTURE ENVIRONMENT
8720
8721 \unitlength = 1pt
8722 \fboxsep = 3pt
8723 \fboxrule = .4pt
8724
8725 %% FOOTNOTES
8726
8727 %\def\footnoterule{} % INITIALIZED BY PLAIN
8728 %\skip\footins{} % INITIALIZED BY PLAIN
8729 %\interfootnotelinepenalty % INITIALIZED BY PLAIN
8730

```

```
8731 \@maxdepth      = \maxdepth
8732
8733 % \vsize initialized because a \clearpage with \vsize < \topskip
8734 % causes trouble.
8735 % \@colroom and \@colht also initialized because \vsize may be
8736 % set to them if a \clearpage is done before the \begin{document}
8737
8738 \vsize = 1000pt
8739 \@colroom = \vsize
8740 \@colht = \vsize
8741
8742 \endinput
```

2.5 report.doc

```

1  % REPORT STANDARD DOCUMENT STYLE -- Released 14 January 1992
2  %   for LaTeX version 2.09
3  % Copyright (C) 1992 by Leslie Lamport
4
5  \typeout{Standard Document Style 'report' <14 Jan 92>}.}
6
7  % PREPARING A FOREIGN LANGUAGE VERSION:
8  %
9  % This document style is for documents prepared in the English language.
10 % To prepare a version for another language, various English words must
11 % be replaced. All the English words that required replacement are
12 % indicated below, where we give the name of the command in which the
13 % words appear, and which must be redefined, with the actual words
14 % underlined.
15 %
16 % \tableofcontents:
17 \def\contentsname{Contents}
18 %
19 %
20 % \listoffigures:
21 \def\listfigurename{List of Figures}
22 %
23 %
24 % \listoftables:
25 \def\listtablename{List of Tables}
26 %
27 %
28 % \thebibliography:
29 \def\bibname{Bibliography}
30 %
31 %
32 % \theindex:
33 \def\indexname{Index}
34 %
35 %
36 % figure environment:
37 \def\figurename{Figure}
38 %
39 %
40 % table environment:
41 \def\tablename{Table}
42 %
43 %
44 % \chapter:
45 \def\chaptername{Chapter}
46 %
47 % \appendix:
48 \def\appendixname{Appendix}
49 %
50 % \part
51 \def\partname{Part}
52 %
53 % abstract environment:
54 \def\abstractname{Abstract}
55 %
56 %
57
58 % CHOOSING THE TYPE SIZE:
59 %
60 % The type size option is handled by reading a different file for each
61 % size, as follows, to define font size-specific commands:
62 %   10pt : REP10, 11pt : REP11, 12pt : REP12
63 % Implemented by \def'ing \@ptsize to last digit of file name.
64 %
65
66 \def\@ptsize{0} % Default is REP10.STY
67 \@namedef{ds@11pt}{\def\@ptsize{1}} % 11pt option reads in REP11.STY
68 \@namedef{ds@12pt}{\def\@ptsize{2}} % 12pt option reads in REP12.STY
69
70 % Two-side or one-side printing.
71 %
72 % \twosidefalse % Default is one-sided printing.

```

```

73 \def\ds@twoside{\@twosidetrue % Defines twoside option.
74         \mparswitchtrue} % Marginpars go on outside of page.
75 % draft option
76 %
77 % \overfullrule = Opt          %Default is don't mark overfull hboxes.
78 \def\ds@draft{\overfullrule 5pt} % Causes overfull hboxes to be marked.
79
80 % FMi 91/03/30: made twocolum.sty a file and twocolumn an option.
81
82 % RmS 91/10/15: moved actual reading of twocolumn.sty
83 %                to the end of this file.
84 %\def\ds@twocolumn{\@input twocolum.sty\relax}
85 \def\ds@twocolumn{\@twocolumntrue}
86
87 % The \@options command causes the execution of every command \ds@FOO
88 % which is defined and for which the user typed the FOO option in his
89 % \documentstyle command. For every option BAR he typed for which
90 % \ds@BAR is not defined, the file BAR.sty will be read after the
91 % present (main) .STY file is executed.
92
93 \@options
94
95 \input rep1\@ptsize.sty\relax
96
97
98 % *****
99 % *                LISTS                *
100 % *****
101 %
102
103 % ENUMERATE
104 % Enumeration is done with four counters: enumi, enumii, enumiii
105 % and enumiv, where enumN controls the numbering of the Nth level
106 % enumeration. The label is generated by the commands \labelenumi
107 % ... \labelenumiv. The expansion of \p@enumN\theenumN defines the
108 % output of a \ref command.
109 %
110 % 16 Mar 88 -- changed defs of \labelenum... to use \theenum...
111
112 \def\labelenumi{\theenumi.}
113 \def\theenumi{\arabic{enumi}}
114
115 \def\labelenumii{(\theenumii)}
116 \def\theenumii{\alph{enumii}}
117 \def\p@enumii{\theenumi}
118
119 \def\labelenumiii{\theenumiii.}
120 \def\theenumiii{\roman{enumiii}}
121 \def\p@enumiii{\theenumi(\theenumii)}
122
123 \def\labelenumiv{\theenumiv.}
124 \def\theenumiv{\Alph{enumiv}}
125 \def\p@enumiv{\p@enumiii\theenumiii}
126
127 % ITEMIZE
128 % Itemization is controlled by four commands: \labelitemi, \labelitemii,
129 % \labelitemiii, and \labelitemiv, which define the labels of the
130 % various itemization levels.
131
132 \def\labelitemi{\m@th\bullet}
133 \def\labelitemii{\bf --}
134 \def\labelitemiii{\m@th\ast}
135 \def\labelitemiv{\m@th\cdot}
136
137
138 % VERSE
139 % The verse environment is defined by making clever use of the
140 % list environment's parameters. The user types \\ to end a line.
141 % This is implemented by \let'in \\ equal \@centercr.
142 %
143 \def\verse{\let\\=\@centercr
144 \list{}{\itemsep\z@ \itemindent -1.5em\listparindent \itemindent
145 \rightmargin\leftmargin\advance\leftmargin 1.5em}\item[]}
146 \let\endverse\endlist

```

```

147
148 % QUOTATION
149 % Fills lines
150 % Indents paragraph
151 %
152 \def\quotation{\list{}{\listparindent 1.5em
153 \itemindent\listparindent
154 \rightmargin\leftmargin\parsep \z@ plus\p@}\item[]}}
155 \let\endquotation=\endlist
156
157 % QUOTE -- same as quotation except no paragraph indentation,
158 %
159 \def\quote{\list{}{\rightmargin\leftmargin}\item[]}}
160 \let\endquote=\endlist
161
162 % DESCRIPTION
163 %
164 % To change the formatting of the label, you must redefine
165 % \descriptionlabel.
166
167 \def\descriptionlabel#1{\hspace\labelsep \bf #1}
168 \def\description{\list{}{\labelwidth\z@ \itemindent-\leftmargin
169 \let\makelabel\descriptionlabel}}
170
171 \let\enddescription\endlist
172
173 \newdimen\descriptionmargin
174 \descriptionmargin=3em
175
176
177 % *****
178 % * OTHER ENVIRONMENTS *
179 % *****
180 %
181 %
182 % THEOREM
183 % \@begintheorem ... \@endtheorem are the commands executed at the
184 % beginning and end of a (user-defined) theorem-like environment.
185 % Except \@opargbegintheorem is executed when an optional argument is
186 % given. Cf. LATEX.TEX.
187 %
188 % \def\@begintheorem#1#2{\it \trivlist
189 % \item[\hskip \labelsep{\bf #1\ #2}]}
190 % \def\@opargbegintheorem#1#2#3{\it \trivlist
191 % \item[\hskip \labelsep{\bf #1\ #2\ (#3)}]}
192 % \def\@endtheorem{\endtrivlist}
193
194
195 % TITLEPAGE
196 % In the normal environments, the titlepage environment does nothing
197 % but start and end a page, and inhibit page numbers. It also resets
198 % the page number to zero. In two-column style, it still makes a
199 % one-column page.
200 \def\titlepage{\@restonecolfalse\if@twocolumn\@restonecoltrue\onecolumn
201 \else \newpage \fi \thispagestyle{empty}\c@page\z@}
202
203 \def\endtitlepage{\if@restonecol\twocolumn \else \newpage \fi}
204
205 % ARRAY AND TABULAR
206 %
207
208 \arraycolsep 5pt % Half the space between columns in an array
209 % environment.
210 \tabcolsep 6pt % Half the space between columns in a tabular
211 % environment.
212 \arrayrulewidth .4pt % Width of rules in array and tabular environment.
213 \doublerulesep 2pt % Space between adjacent rules in array or tabular
214 % environment.
215
216 % TABBING
217 %
218 \tabbingsep \labelsep % Space used by the \' command.
219 % (See LaTeX manual.)
220

```

```

221 % MINIPAGE
222 % \@minipagerestore is called upon entry to a minipage environment to
223 % set up things that are to be handled differently inside a minipage
224 % environment. In the current styles, it does nothing.
225 %
226 % \skip\@mpfootins : plays same role for footnotes in a minipage as
227 % \skip\footins does for ordinary footnotes
228
229 \skip\@mpfootins = \skip\footins
230
231 % FRAMEBOX
232 %
233 \fboxsep = 3pt % Space left between box and text by \fbox and
234 % \framebox.
235 \fboxrule = .4pt % Width of rules in box made by \fbox and \framebox.
236
237
238 % *****
239 % * CHAPTERS AND SECTIONS *
240 % *****
241 %
242 % DEFINE COUNTERS:
243 %
244 % \newcounter{NEWCTR}[OLDCTR] : Defines NEWCTR to be a counter, which is
245 % reset to zero when counter OLDCTR is
246 % stepped.
247 % Counter OLDCTR must already be defined.
248
249 \newcounter {part}
250 \newcounter {chapter}
251 \newcounter {section}[chapter]
252 \newcounter {subsection}[section]
253 \newcounter {subsubsection}[subsection]
254 \newcounter {paragraph}[subsubsection]
255 \newcounter {subparagraph}[paragraph]
256
257 % For any counter CTR, \theCTR is a macro that defines the printed
258 % version of counter CTR. It is defined in terms of the following
259 % macros:
260 %
261 % \arabic{COUNTER} : The value of COUNTER printed as an arabic numeral.
262 % \roman{COUNTER} : Its value printed as a lower-case roman numeral.
263 % \Roman{COUNTER} : Its value printed as an upper-case roman numeral.
264 % \alph{COUNTER} : Value of COUNTER printed as a lower-case letter:
265 % 1 = a, 2 = b, etc.
266 % \Alph{COUNTER} : Value of COUNTER printed as an upper-case letter:
267 % 1 = A, 2 = B, etc.
268 %
269
270 \def\thepart {\Roman{part}}
271 \def\thechapter {\arabic{chapter}}
272 \def\thesection {\thechapter.\arabic{section}}
273 \def\thesubsection {\thesection.\arabic{subsection}}
274 \def\thesubsubsection {\thesubsection.\arabic{subsubsection}}
275 \def\theparagraph {\thesubsubsection.\arabic{paragraph}}
276 \def\thesubparagraph {\theparagraph.\arabic{subparagraph}}
277
278 % \@chapapp is initially defined to be '\chaptername'. The \appendix
279 % command redefines it to be '\appendixname'.
280 %
281 \def\@chapapp{\chaptername}
282
283 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
284 % PART %
285 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
286
287 \def\part{\cleardoublepage % Starts new page.
288 \thispagestyle{plain}% % Page style of part page is 'plain'
289 \if@twocolumn % IF two-column style
290 \onecolumn % THEN \onecolumn
291 \@tempswatrue % @tempswa := true
292 \else \@tempswafalse % ELSE @tempswa := false
293 \fi % FI
294 \hbox{ }\vfil % Add fil glue to center title

```

```

295 %% \bgroup \centering % BEGIN centering %% Removed 19 Jan 88
296 \secdef\@part\@spart}
297
298 \def\@part[#1]#2{\ifnum \c@secnumdepth >-2\relax % IF secnumdepth > -2
299 \refstepcounter{part}% % THEN step
300 % part counter
301 \addcontentsline{toc}{part}{\thepart % add toc line
302 \hspace{1em}#1}\else % ELSE add
303 % unnumb. line
304 \addcontentsline{toc}{part}{#1}\fi % FI
305 \markboth{}{}%
306 {\centering % %% added 19 Jan 88
307 \interlinepenalty \@M %% RmS added 11 Nov 91
308 \ifnum \c@secnumdepth >-2\relax % IF secnumdepth > -2
309 \huge\bf \partname~\thepart % THEN Print '\partname' and
310 \par % number in \huge bold.
311 \vskip 20\p@\fi % Add space before title.
312 \Huge \bf % FI
313 #2\par}\@endpart} % Print Title in \Huge bold.
314 % Bug Fix 13 Nov 89: #1 -> #2
315
316
317 % \@endpart finishes the part page
318 %
319 \def\@endpart{\vfil\newpage % End page with 1fil glue.
320 \if@twoside % IF twoside printing
321 \hbox{}% % THEN Produce totally blank page
322 \thispagestyle{empty}%
323 \newpage
324 \fi % FI
325 \if@tempswa % IF @tempswa = true
326 \twocolumn % THEN \twocolumn
327 \fi} % FI
328
329 \def\@spart#1{\centering % %% added 19 Jan 88
330 \interlinepenalty \@M %% RmS added 11 Nov 91
331 \Huge \bf % Print title in \Huge boldface
332 #1\par}\@endpart}
333
334
335
336
337
338 % *****
339 % * TABLE OF CONTENTS, ETC. *
340 % *****
341 %
342 % A \subsection command writes a
343 % \contentsline{subsection}{TITLE}{PAGE}
344 % command on the .toc file, where TITLE contains the contents of the
345 % entry and PAGE is the page number. If subsections are being numbered,
346 % then TITLE will be of the form
347 % \numberline{NUM}{HEADING}
348 % where NUM is the number produced by \thesubsection. Other sectioning
349 % commands work similarly.
350 %
351 % A \caption command in a 'figure' environment writes
352 % \contentsline{figure}{\numberline{NUM}{CAPTION}}{PAGE}
353 % on the .lof file, where NUM is the number produced by \thefigure and
354 % CAPTION is the figure caption. It works similarly for a 'table'
355 % environment.
356 %
357 % The command \contentsline{NAME} expands to \l@NAME. So, to specify
358 % the table of contents, we must define \l@chapter, \l@section,
359 % \l@subsection, ... ; to specify the list of figures, we must define
360 % \l@figure; and so on. Most of these can be defined with the
361 % \@dottedtocline command, which works as follows.
362 %
363 % \@dottedtocline{LEVEL}{INDENT}{NUMWIDTH}{TITLE}{PAGE}
364 % LEVEL : An entry is produced only if LEVEL < or = value of
365 % 'tocdepth' counter. Note, \chapter is level 0, \section
366 % is level 1, etc.
367 % INDENT : The indentation from the outer left margin of the start
368 % of the contents line.

```

```

369 %   NUMWIDTH : The width of a box in which the section number is to go,
370 %               if TITLE includes a \numberline command.
371 %
372 % This command uses the following three parameters, which are set
373 % with a \def (so em's can be used to make them depend upon the font).
374 % \@pnumwidth : The width of a box in which the page number is put.
375 % \@tocrmarg : The right margin for multiple line entries. One
376 %               wants \@tocrmarg > or = \@pnumwidth
377 % \@dotsep    : Separation between dots, in mu units. Should be
378 %               \def'd to a number like 2 or 1.7
379 %
380 \def\@pnumwidth{1.55em}
381 \def\@tocrmarg {2.55em}
382 \def\@dotsep{4.5}
383 \setcounter{tocdepth}{2}
384
385
386 % TABLEOFCONTENTS
387 % In report style, \tableofcontents, \listoffigures, etc. are always
388 % set in single-column style. @restonecol
389
390 \def\tableofcontents{\@restonecolfalse
391   \if@twocolumn\@restonecoltrue\onecolumn\fi
392   \chapter*{\contentsname
393     \mkboth{\uppercase{\contentsname}}{\uppercase{\contentsname}}}%
394   \@starttoc{toc}\if@restonecol\twocolumn\fi}
395
396 \def\l@part#1#2{\addpenalty{-\@highpenalty}%
397   \addvspace{2.25em plus\p@}% space above part line
398   \begingroup
399   \@tempdima 3em      % width of box holding part number, used by
400   \parindent \z@ \rightskip \@pnumwidth      % \numberline
401   \parfillskip -\@pnumwidth
402   {\large \bf        % set line in \large boldface
403   \leavevmode        % TeX command to enter horizontal mode.
404   #1\hfil \hbox to\@pnumwidth{\hss #2}}\par
405   \nobreak          % Never break after part entry
406   \global\@nobreaktrue      % Added 24 May 89 as
407   \everypar{\global\@nobreakfalse\everypar{}}% suggested by
408   % Jerry Leichter
409   \endgroup}
410
411 %% First line of l@chapter changed 24 May 89, as suggested
412 %% by Jerry Leichter.
413 %%
414 \def\l@chapter#1#2{\addpenalty{-\@highpenalty}%
415   \vskip 1.0em plus\p@ % space above chapter line
416   \@tempdima 1.5em    % width of box holding chapter number
417   \begingroup
418   \parindent \z@ \rightskip \@pnumwidth
419   \parfillskip -\@pnumwidth
420   \bf          % Boldface.
421   \leavevmode  % TeX command to enter horizontal mode.
422   \advance\leftskip\@tempdima %% added 5 Feb 88 to conform to
423   \hskip -\leftskip      % 25 Jan 88 change to \numberline
424   #1\nobreak\hfil \nobreak\hbox to\@pnumwidth{\hss #2}\par
425   \penalty\@highpenalty %% added 24 May 89, suggested by J. Leichter
426   \endgroup}
427
428 \def\l@section{\@dottedtocline{1}{1.5em}{2.3em}}
429 \def\l@subsection{\@dottedtocline{2}{3.8em}{3.2em}}
430 \def\l@subsubsection{\@dottedtocline{3}{7.0em}{4.1em}}
431 \def\l@paragraph{\@dottedtocline{4}{10em}{5em}}
432 \def\l@subparagraph{\@dottedtocline{5}{12em}{6em}}
433
434 % LIST OF FIGURES
435 %
436 \def\listoffigures{\@restonecolfalse
437   \if@twocolumn\@restonecoltrue\onecolumn\fi
438   \chapter*{\listfigurename\mkboth{\uppercase{\listfigurename}}%
439     {\uppercase{\listfigurename}}}\@starttoc{lof}\if@restonecol
440     \twocolumn\fi}
441
442 \def\l@figure{\@dottedtocline{1}{1.5em}{2.3em}}

```



```

443
444 % LIST OF TABLES
445 %
446 \def\listoftables{\@restonecolfalse
447 \if@twocolumn\@restonecoltrue\onecolumn\fi
448 \chapter*{\listtablename\@mkboth{\uppercase{\listtablename}}%
449 {\uppercase{\listtablename}}}\@starttoc{lot}\if@restonecol
450 \twocolumn\fi}
451
452 \let\l@table\l@figure
453
454
455
456 % *****
457 % *          BIBLIOGRAPHY          *
458 % *****
459 %
460 % The thebibliography environment executes the following commands:
461 %
462 % \def\newblock{\hskip .11em plus .33em minus .07em} --
463 %   Defines the 'closed' format, where the blocks (major units of
464 %   information) of an entry run together.
465 %
466 % \sloppy -- Used because it's rather hard to do line breaks in
467 %   bibliographies,
468 %
469 % \sfcode'\.=1000\relax --
470 %   Causes a '.' (period) not to produce an end-of-sentence space.
471
472 %% RmS 91/10/27 [ .. ] replaced by \@biblabel{ .. }
473 %% RmS 91/11/13: Changed counter enumi to enumiv,
474 %%               as it says in the comment in latex.tex
475 %% RmS 92/01/14: Set \p@enumiv to {} and \theenumiv to \arabic{enumiv}
476 %%               to get correct references
477 \def\thebibliography#1{\chapter*{\bibname\@mkboth
478 {\uppercase{\bibname}}{\uppercase{\bibname}}}\list
479 {\@biblabel{\arabic{enumiv}}}{\settowidth\labelwidth{\@biblabel{#1}}%
480 \leftmargin\labelwidth
481 \advance\leftmargin\labelsep
482 \usecounter{enumiv}%
483 \let\p@enumiv\@empty
484 \def\theenumiv{\arabic{enumiv}}}%
485 \def\newblock{\hskip .11em plus .33em minus .07em}%
486 \sloppy\clubpenalty4000\widowpenalty4000
487 \sfcode'\.=\@m}
488
489 %% 91/08/26 FMI & RmS: introduced warning instead of error
490
491 \def\endthebibliography{%
492 \def\@noitemerr{\@warning{Empty 'thebibliography' environment}}%
493 \endlist}
494
495
496 % \def\@biblabel#1{[#1]\hfill} % Produces the label for a \bibitem[...]
497 %                               % command.
498 % \def\@cite#1{[#1]}           % Produces the output of the \cite
499 %                               % command.
500
501
502
503 % *****
504 % *          THE INDEX          *
505 % *****
506 %
507 % THE THEINDEX ENVIRONMENT
508 % Produces double column format, with each paragraph a separate entry.
509 % The user commands \item, \subitem and \subsubitem are used to
510 % produce the entries, and \indexspace adds an extra vertical space
511 % that's the right size to put above the first entry with a new letter
512 % of the alphabet.
513
514 \newif\if@restonecol
515
516 \def\theindex{\@restonecoltrue\if@twocolumn\@restonecolfalse\fi

```

```

517 \columnseprule \z@
518 \columnsep 35\p@\twocolumn[\@makeschapterhead{\indexname}]%
519 \mkboth{\uppercase{\indexname}}{\uppercase{\indexname}}%
520 \thispagestyle{plain}\parindent\z@
521 \parskip\z@ plus .3\p@\relax\let\item\@idxitem}
522
523 \def\@idxitem{\par\hangindent 40\p@}
524
525 \def\subitem{\par\hangindent 40\p@ \hspace*{20\p@}}
526
527 \def\subsubitem{\par\hangindent 40\p@ \hspace*{30\p@}}
528
529 \def\endtheindex{\if@restonecol\onecolumn\else\clearpage\fi}
530
531 \def\indexspace{\par \vskip 10\p@ plus5\p@ minus3\p@\relax}
532
533
534 % *****
535 % * FOOTNOTES *
536 % *****
537 %
538 % \footnoterule is a macro to draw the rule separating the footnotes
539 % from the text. It should take zero vertical space, so it needs a
540 % negative skip to compensate for any positive space taken by the rule.
541 % (See PLAIN.TEX.)
542
543 \def\footnoterule{\kern-3\p@
544 \hrule width .4\columnwidth
545 \kern 2.6\p@} % The \hrule has default height of .4pt.
546
547 % \newcounter{footnote}
548 \@addtoreset{footnote}{chapter} % Numbers footnotes within chapters
549
550 % \@makefnmark :
551 % Must produce the actual footnote, using \@thefnmark as the mark
552 % of the footnote and NOTE as the text. It is called when
553 % effectively inside a \parbox of width \columnwidth (i.e., with
554 % \hsize = \columnwidth).
555 %
556 % The following macro indents all lines of the footnote by 10pt,
557 % and indents the first line of a new paragraph by 1em. To
558 % change these dimensions, just substitute the desired value for
559 % '10pt' [in both places] or '1em'. The mark is flushright
560 % against the footnote.
561 % \long\def\@makefnmark#1{\setpar{\@par\@tempdima \hsize
562 % \advance\@tempdima-10pt\parshape \@ne 10pt \@tempdima}\par
563 % \parindent 1em\noindent
564 % \hbox to \z@{\hss$\m@th{\@thefnmark}$}#1}
565 %
566 % A simpler macro is used, in which the footnote text is
567 % set like an ordinary text paragraph, with no indentation except
568 % on the first line of a paragraph, and the first line of the
569 % footnote. Thus, all the macro must do is set \parindent
570 % to the appropriate value for succeeding paragraphs and put the
571 % proper indentation before mark.
572
573 \long\def\@makefnmark#1{\parindent 1em\noindent
574 \hbox to 1.8em{\hss$\m@th{\@thefnmark}$}#1}
575
576 % \@makefnmark : A macro to generate the footnote marker that goes
577 % in the text. Default used.
578 %
579
580
581
582 % *****
583 % * FIGURES AND TABLES *
584 % *****
585 %
586 % Float placement parameters. See LaTeX manual for their definition.
587 %
588 \setcounter{topnumber}{2}
589 \def\topfraction{.7}
590 \setcounter{bottomnumber}{1}

```

```

591 \def\bottomfraction{.3}
592 \setcounter{totalnumber}{3}
593 \def\textfraction{.2}
594 \def\floatpagefraction{.5}
595 \setcounter{dbltopnumber}{2}
596 \def\dbltopfraction{.7}
597 \def\dblfloatpagefraction{.5}
598
599 % \@makecaption{NUMBER}{TEXT} : Macro to make a figure or table caption.
600 %   NUMBER : Figure or table number--e.g., 'Figure 3.2'
601 %   TEXT   : The caption text.
602 % Macro should be called inside a \parbox of right width, with
603 % \normalsize.
604 % changed 25 Jun 86 to fix according to Howard Trickey:
605 % instead of \unhbox\@tempboxa\par we do #1: #2\par
606
607 \long\def\@makecaption#1#2{%
608   \vskip 10\p@
609   \setbox\@tempboxa\hbox{#1: #2}%
610   \ifdim \wd\@tempboxa >\hsize % IF longer than one line:
611     #1: #2\par                % THEN set as ordinary paragraph.
612   \else                       % ELSE center.
613     \hbox to\hsize{\hfil\box\@tempboxa\hfil}%
614   \fi}
615
616 % To define a float of type TYPE (e.g., TYPE = figure), the document
617 % style must define the following.
618 %
619 % \fps@TYPE : The default placement specifier for floats of type
620 %             TYPE.
621 %
622 % \ftype@TYPE : The type number for floats of type TYPE. Each TYPE
623 %               has associated a unique positive TYPE NUMBER, which
624 %               is a power of two. E.g., figures might have type
625 %               number 1, tables type number 2, programs type number
626 %               4, etc.
627 %
628 % \ext@TYPE : The file extension indicating the file on which the
629 %             contents list for float type TYPE is stored. For
630 %             example, \ext@figure = 'lof'.
631 %
632 % \fnum@TYPE : A macro to generate the figure number for a caption.
633 %             For example, \fnum@TYPE == Figure \thefigure.
634 %
635 % The actual float-making environment commands--e.g., the commands
636 % \figure and \endfigure--are defined in terms of the macros \@float
637 % and \end@float, which are described below.
638 %
639 % \@float{TYPE}[PLACEMENT] : Macro to begin a float environment for a
640 %   single-column float of type TYPE with PLACEMENT as the placement
641 %   specifier. The default value of PLACEMENT is defined by
642 %   \fps@TYPE. The environment is ended by \end@float. E.g.,
643 %   \figure == \@float{figure}, \endfigure == \end@float.
644
645
646 % FIGURE
647
648 \newcounter{figure}[chapter]
649 \def\thefigure{\thechapter.\@arabic\c@figure}
650
651 \def\fps@figure{tbp}
652 \def\ftype@figure{1}
653 \def\ext@figure{lof}
654 \def\fnum@figure{\figurename~\thefigure}
655 \def\figure{\@float{figure}}
656 \let\endfigure\end@float
657 \@namedef{figure*}{\@dblfloat{figure}}
658 \@namedef{endfigure*}{\end@dblfloat}
659
660 % TABLE
661 %
662 \newcounter{table}[chapter]
663 \def\thetable{\thechapter.\@arabic\c@table}
664

```

```

665 \def\fps@table{tbp}
666 \def\ftype@table{2}
667 \def\ext@table{lot}
668 \def\fnun@table{\tablename~\thetable}
669 \def\table{\@float{table}}
670 \let\endtable\end@float
671 \@namedef{table*}{\@dblfloat{table}}
672 \@namedef{endtable*}{\end@dblfloat}
673
674
675 % *****
676 % *      TITLE AND ABSTRACT      *
677 % *****
678 %
679 %
680 % Uses same title and abstract format as the article style's
681 % titlepage option.
682
683 \input titlepag.sty
684
685
686 % *****
687 % *      PAGE STYLES      *
688 % *****
689 %
690 % The page style 'foo' is defined by defining the command \ps@foo. This
691 % command should make only local definitions. There should be no stray
692 % spaces in the definition, since they could lead to mysterious extra
693 % spaces in the output.
694 %
695 % The \ps@... command defines the macros \@oddhead, \@oddfoot,
696 % \@evenhead, and \@evenfoot to define the running heads and
697 % feet---e.g., \@oddhead is the macro to produce the contents of the
698 % heading box for odd-numbered pages. It is called inside an \hbox of
699 % width \textwidth.
700 %
701 % To make headings determined by the sectioning commands, the page style
702 % defines the commands \chaptermark, \sectionmark, ... , where
703 % \chaptermark{TEXT} is called by \chapter to set a mark, and so on.
704 % The \...mark commands and the \...head macros are defined with the
705 % help of the following macros. (All the \...mark commands should be
706 % initialized to no-ops.)
707 %
708 % MARKING CONVENTIONS:
709 % LaTeX extends TeX's \mark facility by producing two kinds of marks
710 % a 'left' and a 'right' mark, using the following commands:
711 %   \markboth{LEFT}{RIGHT} : Adds both marks.
712 %   \markright{RIGHT}      : Adds a 'right' mark.
713 %   \leftmark              : Used in the \@oddhead, \@oddfoot, \@evenhead or
714 %                           \@evenfoot macro, gets the current 'left' mark.
715 %                           Works like TeX's \botmark command.
716 %   \rightmark             : Used in the \@oddhead, \@oddfoot, \@evenhead or
717 %                           \@evenfoot macro, gets the current 'right' mark.
718 %                           Works like TeX's \firstmark command.
719 % The marking commands work reasonably well for right marks 'numbered
720 % within' left marks--e.g., the left mark is changed by a \chapter
721 % command and the right mark is changed by a \section command. However,
722 % it does produce somewhat anomalous results if two \bothmark's occur on
723 % the same page.
724 %
725 %
726 % Commands like \tableofcontents that should set the marks in some
727 % page styles use a \@mkboth command, which is \let by the pagestyle
728 % command (\ps@...) to \markboth for setting the heading or
729 % \@gobbletwo to do nothing.
730
731 \mark{{}{}} % Initializes TeX's marks
732
733 % \ps@empty and \ps@plain defined in LATEX.TEX
734
735 % Definition of 'headings' page style
736 % Note the use of #1 for parameter of \def\chaptermark inside the
737 % \def\ps@headings.
738 %

```

```

739 % 91/03/26 FMi: Added extra set of braces arround |\sl| in |\@oddhead/
740 % to support NFSS (|\sl| is a shape |\rm| a family). Also remove
741 % unnecessary |\hbox{}/ commands.
742 %
743
744 \if@twoside      % If two-sided printing.
745 \def\ps@headings{\let\mkboth\markboth
746 \def\@oddfoot{}\def\@evenfoot{}}% No feet.
747 \def\@oddhead{\rm \thepage\hfil \sl \leftmark}% Left heading.
748 \def\@oddhead{\sl \rightmark}\hfil \rm\thepage}% Right heading.
749 \def\chaptermark##1{\markboth {\uppercase{\ifnum \c@secnumdepth >\m@ne
750 \@chapapp\ \thechapter. \ \fi ##1}}{}}%
751 \def\sectionmark##1{\markright {\uppercase{\ifnum \c@secnumdepth >\z@
752 \thesection. \ \fi ##1}}}}
753 \else          % If one-sided printing.
754 \def\ps@headings{\let\mkboth\markboth
755 \def\@oddfoot{}\def\@evenfoot{}}% No feet.
756 \def\@oddhead{\sl \rightmark}\hfil \rm\thepage}% Heading.
757 \def\chaptermark##1{\markright {\uppercase{\ifnum \c@secnumdepth >\m@ne
758 \@chapapp\ \thechapter. \ \fi ##1}}}}
759 \fi
760
761 % Definition of 'myheadings' page style.
762 %
763 \def\ps@myheadings{\let\mkboth@gobbletwo
764 \def\@oddhead{\sl\rightmark}\hfil \rm\thepage}%
765 \def\@oddfoot{}\def\@evenhead{\rm \thepage\hfil\sl\leftmark}%
766 \def\@evenfoot{}\def\chaptermark##1{}\def\sectionmark##1{}%
767 \def\subsectionmark##1{}}
768
769
770
771 % *****
772 % *      MISCELLANEOUS      *
773 % *****
774 %
775 % DATE
776 %
777 \def\today{\ifcase\month\or
778 January\or February\or March\or April\or May\or June\or
779 July\or August\or September\or October\or November\or December\fi
780 \space\number\day, \number\year}
781
782 % EQUATION and EQNARRAY -- put here because it must follow \chapter
783 % definition
784 %
785 % \newcounter{equation}
786 %
787 \@addtoreset{equation}{chapter} % Makes \chapter reset 'equation'
788 % counter.
789
790 \def\theequation{\thechapter.\arabic{equation}}
791
792 % \jot = 3pt      % Extra space added between lines of an eqnarray
793 % environment
794
795 % The macro \@eqnnum defines how equation numbers are to appear in
796 % equations.
797 %
798 % \def\@eqnnum{(\theequation)}
799 %
800
801
802 % *****
803 % *      INITIALIZATION      *
804 % *****
805 %
806 % Default initializations
807
808 \ps@plain          % 'plain' page style
809 \pagenumbering{arabic} % Arabic page numbers
810 \if@twoside\else\raggedbottom\fi % Ragged bottom unless twoside
811 % option.
812 \if@twocolumn

```

```
813 \@@input twocolum.sty\relax
814 \else
815 \onecolumn           % Single-column.
816 \fi
817
818 \endinput
```

2.6 rep10.doc

```

1  % REPORT STANDARD DOCUMENT STYLE -- Released 14 Jan 1992
2  %   for LaTeX version 2.09
3  % Copyright (C) 1992 by Leslie Lamport
4
5
6  % *****
7  % *                FONTS                *
8  % *****
9  %
10
11 \lineskip 1pt           % \lineskip is 1pt for all font sizes.
12 \normallineskip 1pt
13 \def\baselinestretch{1}
14
15 % Each size-changing command \SIZE executes the command
16 %       \@setsize\SIZE{BASELINESKIP}\FONTSIZE\@FONTSIZE
17 % where:
18 %   BASELINESKIP = Normal value of \baselineskip for that size. (Actual
19 %                   value will be \baselinestretch * BASELINESKIP.)
20 %
21 %   \FONTSIZE    = Name of font-size command. The currently available
22 %                 (preloaded) font sizes are: \vpt (5pt), \vipt (6pt),
23 %                 \viipt (etc.), \iipt, \ixpt, \xpt, \xipt, \xiipt,
24 %                 \xivpt, \xvipt, \xvpt, \xxpt.
25 %   \@FONTSIZE  = The same as the font-size command except with an
26 %                 '0' in front---e.g., if \FONTSIZE = \xivpt then
27 %                 \@FONTSIZE = \@xivpt.
28 %
29 % For reasons of efficiency that needn't concern the designer, the
30 % document style defines \normalsize instead of \normalsize . This
31 % is done only for \normalsize, not for any other size-changing
32 % commands.
33
34 \def\@normalsize{\@setsize\normalsize{12pt}\xpt\xpt
35 \abovedisplayskip 10\p@ plus2\p@ minus5\p@
36 \belowdisplayskip \abovedisplayskip
37 \abovedisplayshortskip \z@ plus3\p@
38 \belowdisplayshortskip 6\p@ plus3\p@ minus3\p@
39 \let\@listi\@listI} % Setting of \@listi added 9 Jun 87
40
41 \def\small{\@setsize\small{11pt}\ixpt\ixpt
42 \abovedisplayskip 8.5\p@ plus3\p@ minus4\p@
43 \belowdisplayskip \abovedisplayskip
44 \abovedisplayshortskip \z@ plus2\p@
45 \belowdisplayshortskip 4\p@ plus2\p@ minus2\p@
46 \def\@listi{\leftmargin\leftmarginI} % Added 22 Dec 87
47 \topsep 4\p@ plus2\p@ minus2\p@\parsep 2\p@ plus\p@ minus\p@
48 \itemsep \parsep}}
49
50 \def\footnotesize{\@setsize\footnotesize{9.5pt}\viipt\@viipt
51 \abovedisplayskip 6\p@ plus2\p@ minus4\p@
52 \belowdisplayskip \abovedisplayskip
53 \abovedisplayshortskip \z@ plus\p@
54 \belowdisplayshortskip 3\p@ plus\p@ minus2\p@
55 \def\@listi{\leftmargin\leftmarginI} % Added 22 Dec 87
56 \topsep 3\p@ plus\p@ minus\p@\parsep 2\p@ plus\p@ minus\p@
57 \itemsep \parsep}}
58
59 \def\scriptsize{\@setsize\scriptsize{8pt}\viipt\@viipt}
60 \def\tiny{\@setsize\tiny{6pt}\vpt\@vpt}
61 \def\large{\@setsize\large{14pt}\xiipt\@xiipt}
62 \def\Large{\@setsize\Large{18pt}\xivpt\@xivpt}
63 \def\LARGE{\@setsize\LARGE{22pt}\xviipt\@xviipt}
64 \def\huge{\@setsize\huge{25pt}\xxpt\@xxpt}
65 \def\Huge{\@setsize\Huge{30pt}\xxvpt\@xxvpt}
66
67 \normalsize % Choose the normalsize font.
68
69
70 % *****
71 % *                PAGE LAYOUT            *
72 % *****

```

```

73 %
74 % All margin dimensions measured from a point one inch from top and side
75 % of page.
76
77 % SIDE MARGINS:
78 \if@twoside           % Values for two-sided printing:
79   \oddsidemargin 44pt % Left margin on odd-numbered pages.
80   \evensidemargin 82pt % Left margin on even-numbered pages.
81   \marginparwidth 107pt % Width of marginal notes.
82 \else                 % Values for one-sided printing:
83   \oddsidemargin 63pt % Note that \oddsidemargin = \evensidemargin
84   \evensidemargin 63pt
85   \marginparwidth 90pt
86 \fi
87 \marginparsep 11pt   % Horizontal space between outer margin and
88                       % marginal note
89
90
91 % VERTICAL SPACING:
92 % Top of page:
93 \topmargin 27pt      % Nominal distance from top of page to top
94                       % of box containing running head.
95 \headheight 12pt    % Height of box containing running head.
96 \headsep 25pt       % Space between running head and text.
97 \topskip = 10pt     % '\baselineskip' for first line of page.
98 % Bottom of page:
99 \footskip 30pt      % Distance from baseline of box containing
100                      % foot to baseline of last line of text.
101
102
103 % DIMENSION OF TEXT:
104 % 24 Jun 86: changed to explicitly compute \textheight to avoid
105 % roundoff. The value of the multiplier was calculated as the floor
106 % of the old \textheight minus \topskip, divided by \baselineskip for
107 % \normalsize. The old value of \textheight was 528pt. \textheight
108 % is the height of text (including footnotes and figures, excluding
109 % running head and foot).
110
111 \textheight = 43\baselineskip
112 \advance\textheight by \topskip
113 \textwidth 345pt     % Width of text line.
114                       % For two-column mode:
115 \columnsep 10pt     % Space between columns
116 \columnseprule 0pt  % Width of rule between columns.
117
118 % A \raggedbottom command causes 'ragged bottom' pages: pages set to
119 % natural height instead of being stretched to exactly \textheight.
120
121 % FOOTNOTES:
122
123 \footnotesep 6.65pt % Height of strut placed at the beginning of every
124                       % footnote = height of normal \footnotesize strut,
125                       % so no extra space between footnotes.
126
127 \skip\footins 9pt plus 4pt minus 2pt % Space between last line of text
128                                       % and top of first footnote.
129
130 % FLOATS: (a float is something like a figure or table)
131 %
132 % FOR FLOATS ON A TEXT PAGE:
133 %
134 % ONE-COLUMN MODE OR SINGLE-COLUMN FLOATS IN TWO-COLUMN MODE:
135 \floatsep 12pt plus 2pt minus 2pt % Space between adjacent floats
136                                       % moved to top or bottom of
137                                       % text page.
138 \textfloatsep 20pt plus 2pt minus 4pt % Space between main text and
139                                       % floats at top or bottom of
140                                       % page.
141 \intextsep 12pt plus 2pt minus 2pt % Space between in-text figures
142                                       % and text.
143 \@maxsep 20pt % The maximum of \floatsep,
144               % \textfloatsep and \intextsep
145               % (minus the stretch and
146               % shrink).

```



```

147 % TWO-COLUMN FLOATS IN TWO-COLUMN MODE:
148 \dblfloatsep 12pt plus 2pt minus 2pt % Same as \floatsep for
149 % double-column figures in
150 % two-column mode.
151 \dbltextfloatsep 20pt plus 2pt minus 4pt % \textfloatsep for
152 % double-column floats.
153 \@dblmaxsep 20pt % The maximum of \dblfloatsep
154 % and \dbltextfloatsep.
155
156 % FOR FLOATS ON A SEPARATE FLOAT PAGE OR COLUMN:
157 % ONE-COLUMN MODE OR SINGLE-COLUMN FLOATS IN TWO-COLUMN MODE:
158 \@fptop Opt plus 1fil % Stretch at top of float page/column. (Must
159 % be Opt plus ...)
160 \@fpsep 8pt plus 2fil % Space between floats on float page/column.
161 \@fpbot Opt plus 1fil % Stretch at bottom of float page/column. (Must
162 % be Opt plus ...)
163
164 % DOUBLE-COLUMN FLOATS IN TWO-COLUMN MODE.
165 \@dblftop Opt plus 1fil % Stretch at top of float page. (Must be Opt
166 % plus ...)
167 \@dblfpsep 8pt plus 2fil % Space between floats on float page.
168 \@dblfpbot Opt plus 1fil % Stretch at bottom of float page. (Must be
169 % Opt plus ...)
170 % MARGINAL NOTES:
171 %
172 \marginparpush 5pt % Minimum vertical separation between two
173 % marginal notes.
174
175
176 % *****
177 % * PARAGRAPHING *
178 % *****
179 %
180 \parskip Opt plus 1pt % Extra vertical space between
181 % paragraphs.
182 \parindent 15pt % Width of paragraph indentation.
183 \topsep 8pt plus 2pt minus 4pt % Extra vertical space, in addition
184 % to \parskip, added above and below
185 % list and paragraphing environments.
186 \partopsep 2pt plus 1pt minus 1pt % Extra vertical space, in addition
187 % to \parskip and \topsep, added when
188 % user leaves blank line before
189 % environment.
190 \itemsep 4pt plus 2pt minus 1pt % Extra vertical space, in addition
191 % to \parskip, added between list
192 % items.
193 % See \@listI for values of \topsep and \itemsep
194 % (Change made 9 Jun 87)
195
196 % The following page-breaking penalties are defined
197
198 \@lowpenalty 51 % Produced by \nopagebreak[1] or \nolinebreak[1]
199 \@medpenalty 151 % Produced by \nopagebreak[2] or \nolinebreak[2]
200 \@highpenalty 301 % Produced by \nopagebreak[3] or \nolinebreak[3]
201
202 \@beginparpenalty -\@lowpenalty % Before a list or paragraph
203 % environment.
204 \@endparpenalty -\@lowpenalty % After a list or paragraph
205 % environment.
206 \@itempenalty -\@lowpenalty % Between list items.
207
208 % \clubpenalty % 'Club line' at bottom of page.
209 % \widowpenalty % 'Widow line' at top of page.
210 % \displaywidowpenalty % Math display widow line.
211 % \predisplaypenalty % Breaking before a math display.
212 % \postdisplaypenalty % Breaking after a math display.
213 % \interlinepenalty % Breaking at a line within a paragraph.
214 % \brokenpenalty % Breaking after a hyphenated line.
215
216
217 % *****
218 % * CHAPTERS AND SECTIONS *
219 % *****
220 %

```

```

221 %
222
223 % Definition of \part moved to report.doc on 19 Jan 88
224
225 % \@makechapterhead {TEXT} : Makes the heading for the \chapter command.
226 %
227
228 \def\@makechapterhead#1{%           % Heading for \chapter command
229   \vspace*{50\p@}%                 % Space at top of text page.
230   {\parindent \z@\raggedright
231     \ifnum \c@secnumdepth >\m@ne % IF secnumdepth > -1 THEN
232       \huge\bf \@chapapp{} \thechapter % Print '\chaptername' and number.
233       \par
234       \vskip 20\p@ \fi             % Space between number and title.
235       \Huge \bf                    % Title.
236       #1\par
237       \nobreak                      % TeX penalty to prevent page break.
238       \vskip 40\p@                 % Space between title and text.
239   }}
240
241 % \@makeschapterhead {TEXT} : Makes the heading for the \chapter*
242 % command.
243 %
244
245 \def\@makeschapterhead#1{%         % Heading for \chapter* command
246   \vspace*{50\p@}%                 % Space at top of page.
247   {\parindent \z@ \raggedright
248     \Huge \bf                       % Title.
249     #1\par
250     \nobreak                        % TeX penalty to prevent page break.
251     \vskip 40\p@                    % Space between title and text.
252   }}
253
254 % \secdef{UNSTARCMDs}{STARCMDs} :
255 %   When defining a \chapter or \section command without using
256 %   \startsection, you can use \secdef as follows:
257 %   \def\chapter { ... \secdef \CMDA \CMDB }
258 %   \def\CMDA  [#1]#2{ ... } % Command to define
259 %   \chapter[...]{...}
260 %   \def\CMDB  #1{ ... } % Command to define
261 %   \chapter*{...}
262
263 \def\chapter{\clearpage % Starts new page.
264   \thispagestyle{plain}% % Page style of chapter page is 'plain'
265   \global\@topnum\z@ % Prevents figures from going
266   % at top of page.
267   \@afterindentfalse % Suppresses indent in first paragraph.
268   \secdef\chapter\schapter} % Change to \@afterindenttrue to
269   % have indent.
270
271 \def\@chapter[#1]#2{\ifnum \c@secnumdepth >\m@ne
272   \refstepcounter{chapter}%
273   \typeout{\@chapapp\space\thechapter.}%
274   \addcontentsline{toc}{chapter}{\protect
275     \numberline{\thechapter}#1}\else
276     \addcontentsline{toc}{chapter}{#1}\fi
277   \chaptermark{#1}%
278   \addtocontents{lof}%
279     {\protect\addvspace{10\p@}} % Adds between-chapter space
280   \addtocontents{lot}%
281     {\protect\addvspace{10\p@}} % to lists of figs & tables.
282   \if@twocolumn % Tests for two-column mode.
283     \@topnewpage[\@makechapterhead{#2}]%
284   \else \@makechapterhead{#2}%
285     \@afterheading % Routine called after chapter and
286     \fi % section heading.
287
288 \def\@schapter#1{\if@twocolumn \@topnewpage[\@makeschapterhead{#1}]%
289   \else \@makeschapterhead{#1}%
290   \afterheading\fi}
291
292 % \@startsection {NAME}-{LEVEL}-{INDENT}-{BEFORESKIP}-{AFTERSKIP}-{STYLE}
293 %   optional * [ALTHEADING]{HEADING}
294 %   Generic command to start a section.

```

```

295 %   NAME       : e.g., 'subsection'
296 %   LEVEL      : a number, denoting depth of section -- e.g.,
297 %               chapter=1, section = 2, etc. A section number will be
298 %               printed if and only if LEVEL < or = the value of the
299 %               secnumdepth counter.
300 %   INDENT     : Indentation of heading from left margin
301 %   BEFORESKIP : Absolute value = skip to leave above the heading.
302 %               If negative, then paragraph indent of text following
303 %               heading is suppressed.
304 %   AFTERSKIP  : if positive, then skip to leave below heading,
305 %               else - skip to leave to right of run-in heading.
306 %   STYLE      : commands to set style
307 %   If '*' missing, then increments the counter. If it is present, then
308 %   there should be no [ALTHEADING] argument. A sectioning command
309 %   is normally defined to \@startsection + its first six arguments.
310
311 \def\section{\@startsection {section}{1}{\z@}{-3.5ex plus-1ex minus
312 - .2ex}{2.3ex plus.2ex}{\reset@font\Large\bf}}
313 \def\subsection{\@startsection{subsection}{2}{\z@}{-3.25ex plus-1ex
314 minus-.2ex}{1.5ex plus.2ex}{\reset@font\large\bf}}
315 \def\subsubsection{\@startsection{subsubsection}{3}{\z@}{-3.25ex plus
316 -1ex minus-.2ex}{1.5ex plus.2ex}{\reset@font\normalsize\bf}}
317 \def\paragraph{\@startsection
318 {paragraph}{4}{\z@}{3.25ex plus1ex minus.2ex}{-1em}{\reset@font
319 \normalsize \bf}}
320 \def\subparagraph{\@startsection
321 {subparagraph}{4}{\parindent}{3.25ex plus1ex minus
322 .2ex}{-1em}{\reset@font\normalsize\bf}}
323
324
325 % Default initializations of \..mark commands. (See below for their
326 % us in defining page styles.
327 %
328
329 \def\chaptermark#1{}
330 % \def\sectionmark#1{}           % Preloaded definitions
331 % \def\subsectionmark#1{}
332 % \def\subsubsectionmark#1{}
333 % \def\paragraphmark#1{}
334 % \def\subparagraphmark#1{}
335
336 % The value of the counter secnumdepth gives the depth of the
337 % highest-level sectioning command that is to produce section numbers.
338 %
339
340 \setcounter{secnumdepth}{2}
341
342 % APPENDIX
343 %
344 % The \appendix command must do the following:
345 % -- reset the chapter counter to zero
346 % -- set \@chapapp to Appendix (for messages)
347 % -- redefine the chapter counter to produce appendix numbers
348 % -- reset the section counter to zero
349 % -- redefine the \chapter command if appendix titles and headings
350 % are to look different from chapter titles and headings.
351
352 \def\appendix{\par
353 \setcounter{chapter}{0}%
354 \setcounter{section}{0}%
355 \def\@chapapp{\appendixname}%
356 \def\thechapter{\Alph{chapter}}}
357
358
359 % *****
360 % *           LISTS           *
361 % *****
362 %
363
364 % The following commands are used to set the default values for the list
365 % environment's parameters. See the LaTeX manual for an explanation of
366 % the meanings of the parameters. Defaults for the list environment are
367 % set as follows. First, \rightmargin, \listparindent and \itemindent
368 % are set to Opt. Then, for a Kth level list, the command \@listK is

```

```

369 % called, where 'K' denotes 'i', 'ii', ... , 'vi'. (I.e., \@listiii is
370 % called for a third-level list.) By convention, \@listK should set
371 % \leftmargin to \leftmarginK.
372 %
373 % For efficiency, level-one list's values are defined at top level, and
374 % \@listi is defined to set only \leftmargin.
375
376 \leftmargini 25pt
377 \leftmarginii 22pt % > \labelsep + width of '(m)'
378 \leftmarginiii 18.7pt % > \labelsep + width of 'vii.'
379 \leftmarginiv 17pt % > \labelsep + width of 'M.'
380 \leftmarginv 10pt
381 \leftmarginvi 10pt
382
383 \leftmargin\leftmargini
384 \labelsep 5pt
385 \labelwidth\leftmargini\advance\labelwidth-\labelsep
386 %\parsep 4pt plus 2pt minus 1pt (Removed 9 Jun 87)
387
388 % \@listI defines top level and \@listi values of
389 % \leftmargin, \topsep, \parsep, and \itemsep
390 % (Added 9 Jun 87)
391 \def\@listI{\leftmargin\leftmargini \parsep 4\p@ plus2\p@ minus\p@%
392 \topsep 8\p@ plus2\p@ minus4\p@
393 \itemsep 4\p@ plus2\p@ minus\p@}
394
395 \let\@listi\@listI
396 \@listi
397
398 \def\@listii{\leftmargin\leftmarginii
399 \labelwidth\leftmarginii\advance\labelwidth-\labelsep
400 \topsep 4\p@ plus2\p@ minus\p@
401 \parsep 2\p@ plus\p@ minus\p@
402 \itemsep \parsep}
403
404 \def\@listiii{\leftmargin\leftmarginiii
405 \labelwidth\leftmarginiii\advance\labelwidth-\labelsep
406 \topsep 2\p@ plus\p@ minus\p@
407 \parsep \z@ \partopsep\p@ plus\z@ minus\p@
408 \itemsep \topsep}
409
410 \def\@listiv{\leftmargin\leftmarginiv
411 \labelwidth\leftmarginiv\advance\labelwidth-\labelsep}
412
413 \def\@listv{\leftmargin\leftmarginv
414 \labelwidth\leftmarginv\advance\labelwidth-\labelsep}
415
416 \def\@listvi{\leftmargin\leftmarginvi
417 \labelwidth\leftmarginvi\advance\labelwidth-\labelsep}
418
419 \endinput

```

第3章 L^AT_EX 2_ε

文書スタイルのカスタマイズ

〔レスリー・ランボート（阿瀬はる美訳）『文書処理システム L^AT_EX 2_ε』（ピアソン・エデュケーション・1999年）103–105頁〕

L^AT_EX の標準スタイルオプションによって生成される出力のスタイルが気に入らないときは、希望に沿った出力を生成してくれるオプションがないかどうか『The L^AT_EX Companion』やローカルガイドを調べてみてほしい。望むようなオプションが用意されていない場合は、文書のスタイルを自分自身で変更しなければならない。ここでいう“文書スタイルの変更”とは、段落やリストといった通常の構造の出力形式を変えることであり、新たな構造を作成するという意味ではない。新しい論理構造を作成する方法については、3.4節を参照のこと。

文書スタイルのカスタマイズに取り組む前に、自分で文書をデザインしようとした作者の多くが、ごく初歩的な誤りを犯しているという事実を知っておいてほしい。彼らのような誤りを避けるためには、熟練した組版デザイナーに相談するか、あるいは組版デザインに関する書籍などを読んで自分で勉強するしかない。ただ、ここでは、犯しがちな間違いを1つだけ指摘しておこう。それは、1行が長すぎると文が非常に読みづらくなる、ということである。そこで筆者は、“1行の長さは、句読点やスペース文字も含めて75文字以内にすること”という提案をしておくことにする〔訳注（阿瀬）：これは欧文の場合の指標である。文書の版形にもよるが、和文の場合であれば1行の文字数は40字前後までが適当で、これを超えると読みづらくなるようである。〕。

特定の文書のスタイルを変更する場合であれば、その文書のプリアンプルに宣言を付け加えるだけでよい。しかし、いくつかの文書について同じスタイル変更を行う場合は、宣言を1つのパッケージにまとめておいたほうが効率的である。パッケージは、必要な宣言を sty ファイル中に書き入れてやれば作成できる。sty ファイルとは、主ファイル名がパッケージ名、拡張子が sty のファイルのことである。たとえば、vacation という名前のパッケージを定義するときは、vacation.sty というファイルを作成すればよい。ファイルを作成したら、`\usepackage{vacation}` というコマンドを使えば、vacation.sty ファイルを L^AT_EX に読み込ませることができる。

なお、パッケージの sty ファイルを読み込む際、T_EX は@（アット記号）をふつうの英文字として扱うことに注意してほしい。このため、`\@listi` のようにコマンド名の中で@記号を使うことができる。しかし、文書中では、このようなコマンド名を使うことはできない。T_EX は、`\@` というコマンドの後に listi という文字列があると解釈してしまうからである。L^AT_EX の内部コマンドの多くには、文書中で誤って使われるのを防ぐために、あまり使われることのない@記号が含まれている。文書スタイルを変更する際に使われるパラメータにも、@記号が含まれているものがある（付録Cを参照）。

文書のスタイルを変更する最も手軽な方法は、ページ

に出力されるテキストの高さや幅などを決めるパラメータを変えることだろう。L^AT_EX のスタイルパラメータに関しては、この章および付録Cで説明している。パラメータを変更しただけでは望むような結果が得られないときは、L^AT_EX コマンドを再定義する必要がある。たとえば、章の見出し形式を変更したいときなどは、`\chapter` コマンドを再定義しなければならない。さらに、章の見出しに以前の章の表題をすべて書き出したいなど、より複雑な作業が必要になる場合には、高度な T_EX のコマンドを勉強しなくてはならない。しかし、そのような複雑な作業を必要とする変更はめったにないと思われる。ここでは、スタイル変更のやりかたを示すために、典型的なスタイル変更の例を1つ挙げ、それについて説明する。具体的には、章の表題を通常のローマンのボールド体ではなく、サンセリフのボールド体で印字するようにしてみよう。

`\chapter` の定義を変更することになるので、まず `\chapter` の定義がどこにあるかを調べる必要がある。すべての文書クラスに共通するコマンドは、プリロード（事前にロード）されていることが多い。プリロードされたコマンドが定義されているファイルの名前は、`source2e.tex` というファイルに書かれている（このファイルがどのディレクトリにあるかについては、ローカルガイドを参照のこと）。しかし、`\chapter` のように、特定の文書クラスにしか存在しないコマンドやパッケージによって定義されているコマンドは、ふつうプリロードされていない。これらのコマンドの定義は、文書を処理するときに L^AT_EX が読み込むファイルに収められている（L^AT_EX が読み込むファイルについては、4.7節を参照）。

ここでは、report 文書クラスを使って文書を作成しているものとする。この場合、L^AT_EX は `report.cls` というファイルを読み込む。そこで、このファイルの中から“`\chapter`”という文字列を探せば、次のような行が見つかるはずだ。

```
\newcommand\chapter{\if@openright\cleardoublepage\else...
  \thispagestyle{plain}%
  \global\@topnum\z@
  \@afterindentfalse
  \secdef\@chapter\@schapter}
```

`\chapter` コマンドは、ここで定義されている（T_EX プログラミングのエキスパートは、`\newcommand` の最初の引数を囲んでいるプレースを省略してしまうことがあるが、これは真似しない方がよい）。この定義を見ればすぐに、章の見出しを生成するのは `\@chapter`、あるいは `\@schapter` であろうと見当がつく。`\chapter` の定義のすぐ下には、次のような行がある。

```
\def\@chapter[#1]#2{...
```

`\@chapter` は、ここで定義されているということだ（T_EX プログラミングのエキスパートは、T_EX コマンドであ

\def を使ってコマンドを定義することがあるが、これも決して真似しないでほしい)。この定義を見ると、見出しは \makechapterhead コマンドによって生成されると推測できる。さらに \makechapterhead コマンドの定義では、2つの \bfseries コマンドが使われている。これら2つのコマンドの前か後に \sffamily コマンドを付け加えれば、きっと章の見出しをサンセリフのボールド体で生成することができるに違いない。そこで、sfchap.sty という名前のファイルを作り、そこに変更した \makechapterhead の定義を収める。そして、文書のプリアンプルに \usepackage{sfchap} コマンドを付け加えてから、L^AT_EX を実行してみよう。すると、L^AT_EX は、希望どおり、サンセリフのボールド体で見出しを出力してくれる。最初の推理は正しかったのである！

以上の例は、スタイルを変更するときのためのよいヒントになるだろう。しかし、もっと簡単にスタイルを変更する方法もある。report.cls の先頭にある注釈を読んでみると、report.cls ファイルは classes.dtx という名前のソースファイルから生成されたことがわかる (source2e.tex にも、classes.dtx のことが書かれている)。dtx ファイルには、注釈と整形コマンドが入っている。L^AT_EX で処理すれば dtx ファイルを紙に印刷

できるが、dtx ファイル自身をそのまま読んでもよい。classes.dtx からは、\schapter コマンドが、番号付けされていない章の見出しを生成するということがわかる (番号付けされていない見出しは、\chapter* コマンドで生成できる。C.4.1 節を参照)。ここで注釈を読めば、番号付けされていない章見出しの出力形式を変更する方法がわかるはずである。

以上、コマンドの変更のしかたについて説明したが、環境の変更も同様にして行うことができる。ただし、環境によっては \newenvironment ではなく、\def という T_EX のコマンドで定義されているものもあるので、定義の書かれている場所を探すときには注意が必要である。たとえば、equation 環境は、\begin{equation} によって実行される \equation と \end{equation} によって実行される \endequation という、2つのコマンドによって定義されている。

スタイル変更のためのパッケージを自分で作成しようと思いついたときには、何よりもまず『The L^AT_EX Companion』やローカルガイドなどを読み、すでにだれかが自分の希望するような出力結果を生成してくれる文書クラスやパッケージを作成していないかどうか、調べるようにすべきだろう。

Plain T_EX コマンドを使う〔同書 266–267 頁〕

L^AT_EX は、T_EX の“マクロパッケージ”，つまり、あらかじめ定義された T_EX コマンドの集まりとして実現されている。Plain T_EX は、“裸”の T_EX とマクロパッケージ plain によって構成された、T_EX の標準版である。Plain T_EX コマンドを使えば、標準の L^AT_EX コマンドではできないようなことを実現できる。ただし、Plain T_EX を使おうと考える前に、自分が希望する処理を行ってくれるようなパッケージがないかどうか、『The L^AT_EX Companion』を読んで調べるようにしたほうがよいだろう。

ほとんどの Plain T_EX コマンドは、L^AT_EX の中でも使うことができるが、使用に際しては注意が必要である。L^AT_EX は、その中のコマンド群が集まって1つのシステムとして機能するように設計されている。L^AT_EX の各コマンドは、ほかのコマンドといっしょに使ったときにもうまく機能するよう、さまざまな妥協がなされているのである。したがって、L^AT_EX コマンドを、本書で説明されていない Plain T_EX コマンドと組み合わせて使った場合、うまく機能しないかもしれない。

Plain T_EX コマンドが問題を引き起こすかどうかは、簡単に判断することはできない。これは、実際に試してみるしか方法はない。一般則としては、L^AT_EX のコマンドや環境が利用しているパラメータを変更する可能性のある Plain T_EX のコマンドを、そのコマンドや環境と組み合わせて使うのは避けたほうがよい。たとえば、L^AT_EX のリスト作成環境中では、T_EX の段落作成パラメータを変更する \hangindent のような Plain T_EX コマンドを使ってはならない。

L^AT_EX の \output ルーチンで使われるパラメータについては、本書中で指定されている以外の方法で変更してはならない。とくに『The T_EXbook』の15章については、そのほとんどはすでに適用できないと考えたほうがよい。とはいえ、L^AT_EX はレジスタ割り当てに関する T_EX の規約をすべて受け継いでいるため、通常の T_EX コマンドを使って独自のカウンタやボックスなどを定義

することは可能である。

L^AT_EX の中で定義が削除、あるいは変更されている Plain T_EX コマンドは、次に示すとおりである。L^AT_EX コマンドの中で Plain T_EX コマンドで似たような機能を持つものや、名前の中に @ が含まれる“内部”コマンドについては、ここには示していない。

タブ・コマンド

L^AT_EX には tabbing 環境があるため、次のコマンドは削除されている。

```
\tabs      \tabsdone   \settabs   \+
\tabset    \cleartabs \tabalign
```

出力、脚注、図

次のコマンドは、Plain T_EX の出力ルーチンを必要とするため、削除されている。これらのコマンドは、L^AT_EX の脚注作成コマンド、figure 環境、あるいは table 環境によって置き換えられている。

```
\pageno      \nopagenumbers  \makeheadline  \topins
\headline    \advancepageno  \makefootline  \topinsert
\footline    \pagebody      \dosupereject  \midinsert
\normalbottom \plainoutput    \footstrut     \pageinsert
\folio       \pagecounters   \vfootnote     \endinsert
```

フォント選択コマンド

次の Plain T_EX コマンドは、L^AT_EX では定義されていない。

```
\fivei      \fivebf      \seveny
\fivevm     \seveni      \teni
\fivesy     \sevenbf     \oldstyle
```

数式の位置揃え

L^AT_EX には eqnarray 環境や eqnarray* 環境があるため、次の Plain T_EX コマンドは削除されている。

```
\equalign   \equalignno   \lequalignno
```

そのほか

Plain TeX の \$\$ は正しく機能しないが、L^AT_EX コマンドでは \[と \] に置き換えられている。Plain TeX の \beginsection コマンドは、L^AT_EX のセクションコマンドに置き換えられている。また、\end と \bye コマンドは、\end{document} に置き換えられている。Plain TeX コマンドの \centering と \line は、L^AT_EX では

同名の別機能のコマンドとなっており、\input コマンドの構文も L^AT_EX の規約に従うように変更されている。Plain TeX の \line コマンドで行えることはほとんど、center, flushleft, あるいは flushright の各環境を利用すれば実現できる。なお、L^AT_EX には、Plain TeX の \magnification コマンドに対応するコマンドはないが、出力の拡大は、dvi ファイルの印字プログラムによって行える場合が多い。

source2e.tex

ランポートさんの本にも出て来る “source2e.tex” を以下に挙げておきます（このファイル自体はどなたの PC の中にもあるはずです）。\end{document} の後ろ（341 行以下）に書いてある指示に従ってタイプセットすると、“The L^AT_EX 2_ε Sources” という約 550 ページのドキュメントになります。216 行～296 行で “\DocInclude” している 41 個の dtx ファイルをひとつにまとめて、目次やコマンドインデックスを付けたものです（202 行以下にも書かれていますように、必要な dtx ファイルだけを個々にタイプセットしてももちろん構いません。でも、まとめてタイプセットして、コマンドインデックスも作っておくと、検索するのに重宝します）。これが、“kernel” と呼ばれる L^AT_EX の中核部分のコードと、それについての説明ですね。source2e.tex から説明の部分を取り去ったものが、“latex.ltx”（kernel そのもの）なんだと思います。

この『別冊付録』では、コードをカラフルにしたかったというだけの理由で listings パッケージを使っていますが、doc パッケージを使うと、source2e.tex と同様に、コマンドインデックスを作ることが出来ます（source2e.tex で使われているクラスファイル “ltxdoc.cls” は、内部で doc パッケージを読み込んでいます）。ですんで、“plain.tex” や “latex.tex” なんかも、doc パッケージを使えば、コマンドインデックスを得ることが出来ます。

あと、hypdoc というパッケージもあって、こちらを使うと doc パッケージだけでなく hyperref パッケージも読み込んでタイプセット出来るので、そうすると、目次やコマンドインデックスにリンクが張られた pdf が作れて、検索がもっと楽チンになります。

```

1 % \iffalse meta-comment
2 %
3 % Copyright 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006
4 % The LaTeX3 Project and any individual authors listed elsewhere
5 % in this file.
6 %
7 % This file is part of the LaTeX base system.
8 % -----
9 %
10 % It may be distributed and/or modified under the
11 % conditions of the LaTeX Project Public License, either version 1.3c
12 % of this license or (at your option) any later version.
13 % The latest version of this license is in
14 %   http://www.latex-project.org/lppl.txt
15 % and version 1.3c or later is part of all distributions of LaTeX
16 % version 2005/12/01 or later.
17 %
18 % This file has the LPPL maintenance status "maintained".
19 %
20 % The list of all files belonging to the LaTeX base distribution is
21 % given in the file 'manifest.txt'. See also 'legal.txt' for additional
22 % information.
23 %
24 % The list of derived (unpacked) files belonging to the distribution
25 % and covered by LPPL is defined by the unpacking scripts (with
26 % extension .ins) which are part of the distribution.
27 %
28 % \fi
29
30 % This document will typeset the LaTeX sources as a single document.
31 % This will produce quite a large file (roughly 555 pages) and may

```

```

32 % take a long time.
33 %
34 % Some notes on processing this document are contained at the end
35 % of this document, after \end{document}
36 %
37 % DPC 1997/11/17
38 %
39 % First a special index style for makeindex
40 %
41
42 \begin{filecontents}{source2e.ist}
43 actual '='
44 quote '!'
45 level '>'
46 preamble
47 "\n \begin{theindex} \n \makeatletter\scan@allowedfalse\n"
48 postamble
49 "\n\n \end{theindex}\n"
50 item_x1 "\\efill \n \subitem "
51 item_x2 "\\efill \n \subsubitem "
52 delim_0 "\\pfill "
53 delim_1 "\\pfill "
54 delim_2 "\\pfill "
55 % The next lines will produce some warnings when
56 % running Makeindex as they try to cover two different
57 % versions of the program:
58 lethead_prefix "{\\bfseries\\hfil "
59 lethead_suffix "\\hfil}\\nopagebreak\n"
60 lethead_flag 1
61 heading_prefix "{\\bfseries\\hfil "
62 heading_suffix "\\hfil}\\nopagebreak\n"
63 headings_flag 1
64
65 % and just for source2e:
66 % Remove R so I is treated in sequence I J K not I II III
67 page_precedence "rna"
68 \end{filecontents}
69
70
71
72 \documentclass{ltxdoc}
73
74 \listfiles
75
76 % Do not index some TeX primitives, and some common plain TeX commands.
77
78 \DoNotIndex{\def,\long,\edef,\xdef,\gdef,\let,\global}
79 \DoNotIndex{\if,\ifnum,\ifdim,\ifcat,\ifmmode,\ifvmode,\ifhmode,%
80 \iftrue,\iffalse,\ifvoid,\ifx,\ifeof,\ifcase,\else,\or,\fi}
81 \DoNotIndex{\box,\copy,\setbox,\unvbox,\unhbox,\hbox,%
82 \vbox,\vtop,\vcenter}
83 \DoNotIndex{\@empty,\immediate,\write}
84 \DoNotIndex{\egroup,\bgroup,\expandafter,\begingroup,\endgroup}
85 \DoNotIndex{\divide,\advance,\multiply,\count,\dimen}
86 \DoNotIndex{\relax,\space,\string}
87 \DoNotIndex{\csname,\endcsname,\@spaces,\openin,\openout,%
88 \closein,\closeout}
89 \DoNotIndex{\catcode,\endinput}
90 \DoNotIndex{\jobname,\message,\read,\the,\m@ne,\noexpand}
91 \DoNotIndex{\hsize,\vsize,\hskip,\vskip,\kern,\hfil,\hfill,\hss}
92 \DoNotIndex{\m@ne,\z@,\z@skip,\@ne,\tw@,\p@}
93 \DoNotIndex{\dp,\wd,\ht,\vss,\unskip}
94
95 % Set up the Index and Change History to use \part
96 \IndexPrologue{\part*{Index}%
97 \markboth{Index}{Index}%
98 \addcontentsline{toc}{part}{Index}%
99 The italic numbers denote the pages where the
100 corresponding entry is described,
101 numbers underlined point to the definition,
102 all others indicate the places where it is used.}
103
104 \GlossaryPrologue{\part*{Change History}%
105 % Allow control names to be hyphenated here...

```



```

106         {\GlossaryParms\ttfamily\hyphenchar\font='{-}%
107         \markboth{Change History}{Change History}%
108         \addcontentsline{toc}{part}{Change History}}
109
110 % The standard \changes command modified slightly to better cope with
111 % this multiple file document.
112 \makeatletter
113 \def\changes@#1#2#3{%
114   \let\protect\@unexpandable@protect
115   \edef\@tempa{\noexpand\glossary{#2\space\currentfile\space#1\levelchar
116     \ifx\saved@macroname\@empty
117       \space
118       \actualchar
119       \generalname
120     \else
121       \expandafter\@gobble
122       \saved@macroname
123       \actualchar
124       \string\verb\quotechar*%
125       \verbatimchar\saved@macroname
126       \verbatimchar
127     \fi
128     :\levelchar #3}}%
129   \@tempa\endgroup\@esphack}
130 \makeatother
131
132 % Produce a Change Log and (2 column) Index.
133 \RecordChanges
134 \CodelineIndex
135 \EnableCrossrefs
136 \setcounter{IndexColumns}{2}
137
138 % Needed for documentation in ltoutenc.dtx
139 \usepackage{textcomp}
140
141 \begin{document}
142 \title{The \LaTeXe\ Sources}
143 \author{%
144   Johannes Braams\\
145   David Carlisle\\
146   Alan Jeffrey\\
147   Leslie Lamport\\
148   Frank Mittelbach\\
149   Chris Rowley\\
150   Rainer Sch\"opf}
151
152 % This command will be used to input the patch file
153 % if that file exists.
154 \newcommand{\includeltpatch}{%
155   \def\currentfile{ltpatch.ltx}
156   \part{ltpatch}
157   {\let\ttfamily\relax
158     \xdef\filekey{\filekey, \thepart={\ttfamily\currentfile}}}%
159   Things we did wrong\ldots
160   \IndexInput{ltpatch.ltx}}
161
162
163
164 % Get the date from ltvers.dtx
165 \makeatletter
166 \let\patchdate=\@empty
167 \begingroup
168   \def\ProvidesFile#1\fmtversion#2{\date{#2}\endinput}
169   \input{ltvers.dtx}
170 \global\let\X@date=\@date
171
172 % Add the patch version if available.
173 \long\def\Xdef#1#2#3\def#4#5{%
174   \xdef\X@date{#2}%
175   \xdef\patchdate{#5}%
176   \endinput}%
177 \InputIfFileExists{ltpatch.ltx}
178 {\let\def\Xdef}{\global\let\includeltpatch\relax}
179 \endgroup

```

```

180
181 \ifx\@date\X@date
182   \def\Xpatch{0}
183   \ifx\patchdate\Xpatch\else
184     \edef\@date{\@date\space Patch level \patchdate}
185   \fi
186 \else
187   \@warning{ltpatch.ltx does not match ltvers.dtx!}
188   \let\includeltpatch\relax
189 \fi
190 \makeatother
191
192 \pagenumbering{roman}
193 \maketitle
194 \renewcommand\maketitle{}
195
196 \tableofcontents
197
198 \clearpage
199
200 \pagenumbering{arabic}
201
202 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
203
204 % Each of the following \DocInclude lines includes a file with extension
205 % .dtx. Each of these files may be typeset separately. For instance
206 % latex ltboxes.dtx
207 % will typeset the source of the LaTeX box commands.
208 %
209 % If this file is processed, each of these separate dtx files will be
210 % contained as a part of a single document. Using ltadoc.cfg you can
211 % then optionally produce a combined index and/or change history for
212 % the entire source of the format file. Note that such a document will
213 % be quite large (about 555 pages).
214 %
215
216 \DocInclude{ltldirchk} % System dependant initialisation
217
218 \DocInclude{ltplain} % LaTeX version of Knuth's plain.tex
219
220 \DocInclude{ltvers} % Current version date
221
222 \DocInclude{ltdefns} % Initial definitions.
223
224 \DocInclude{ltalloc} % Allocation of counters and others.
225
226 \DocInclude{ltctrl} % Program control macros.
227
228 \DocInclude{lterror} % Error handling.
229
230 \DocInclude{ltpar} % Paragraphs.
231
232 \DocInclude{ltspc} % Spacing, line and page breaking.
233
234 \DocInclude{ltlogos} % Logos.
235
236 \DocInclude{ltfiles} % \input files and related commands
237
238 \DocInclude{ltoutenc} % Output encoding interface
239
240 \DocInclude{ltcounts} % Counters
241
242 \DocInclude{ltlength} % Lengths
243
244 \DocInclude{ltfssbas} % NFSS Base macros
245
246 \DocInclude{ltfssstrc} % NFSS Tracing (and tracefmt.sty)
247
248 \DocInclude{ltfsscmp} % NFSS1 Compatibility
249
250 \DocInclude{ltfssdcl} % NFSS Declarative interface
251
252 \DocInclude{ltfssini} % NFSS Initialisation
253

```

```

254 \DocInclude{fontdef} % fonttext.ltx/fontmath.ltx
255
256 \DocInclude{preload} % preload.ltx
257
258 \DocInclude{ltfntcmd} % \textrm etc
259
260 \DocInclude{ltpageno} % Page numbering
261
262 \DocInclude{ltxref} % Cross referencing
263
264 \DocInclude{ltmiscen} % Miscellaneous environment definitions.
265
266 \DocInclude{ltmath} % Mathematics set up.
267
268 \DocInclude{ltlists} % List and related environments
269
270 \DocInclude{ltboxes} % Parbox and friends
271
272 \DocInclude{lftab} % Tabbing tabular and array
273
274 \DocInclude{ltpictur} % Picture mode
275
276 \DocInclude{ltthm} % Theorem environments
277
278 \DocInclude{ltsect} % Sectioning
279
280 \DocInclude{ltfloat} % Floats
281
282 \DocInclude{ltidxglo} % Index and Glossary
283
284 \DocInclude{ltbibl} % Bibliography
285
286 \DocInclude{ltpage} % \pagestyle \raggedbottom \sloppy
287
288 \DocInclude{ltoutput} % Output routine
289
290 \DocInclude{ltclass} % Package & Class interface
291
292 \DocInclude{lthyphen} % Hyphenation (hyphen.ltx).
293
294 \DocInclude{ltfinal} % Last minute initialisations and dump
295
296 \includelpatch % Corrections distributed after the full release
297
298 % Stop here if ltxdoc.cfg says \AtEndOfClass{\OnlyDescription}
299 \StopEventually{\end{document}}
300
301 \clearpage
302 \pagestyle{headings}
303
304 % Make TeX shut up.
305 \hbadness=10000
306 \newcount\hbadness
307 \hfuzz=\maxdimen
308
309 \typeout{%
310 \string # Produce change log with^^J%
311 makeindex -s gglo.ist -o source2e.gls source2e.glo}
312
313
314 \PrintChanges
315
316 \clearpage
317
318 % makeindex needs a symbol between the parts of composite page numbers
319 % but we dont want one, so:
320 \typeout{%
321 \string # Produce index with^^J%
322 makeindex -s source2e.ist source2e.idx}
323
324 \begingroup
325 \def\endash{--}
326 \catcode'\-\active
327 \def-{\futurelet\temp\indexdash}

```

```

328 \def\indexdash{\ifx\temp-\endash\fi}
329
330 \PrintIndex
331 \endgroup
332
333 % Make sure that the index is not printed twice
334 % (ltxdoc.cfg might have a second \PrintIndex command)
335 \let\PrintChanges\relax
336 \let\PrintIndex\relax
337
338 \end{document}
339
340
341 %%%%%%%%%%%
342
343 To use this file to produce a fully indexed source code
344 you need to execute the following (or equivalent) commands:
345
346     latex source2e.tex
347
348     makeindex -s source2e.ist source2e.idx
349     makeindex -s gglo.ist -o source2e.gls source2e.glo
350
351     latex source2e.tex
352     latex source2e.tex
353
354
355 The makeindex style source2e.ist is used in place of the usual
356 doc gind.ist to ensure that I is used in the sequence I J K
357 not I II II, which would be the default makeindex behaviour.
358
359 The third run with latex is only required to get the table of
360 contents entries for the change log and index. You may speed things up
361 by using the \includeonly mechanism so as not to typeset the source
362 files on the second run. This involves changing the file
363 ltxdoc.cfg
364 between the latex runs.
365
366 The following unix script automates this.
367 (It could easily be ported to DOS or VMS,
368  rm is ReMove a file, and echo "... " > file writes ... to "file".)
369
370
371 After this script (after the second =====) is a similar script
372 that will produce the documentation for all the files in the base
373 distribution that are *not* included in source2e.dvi. This second script
374 was requested, but before using it, beware it will take a long time!
375 It may however be modified as required, eg to not typeset the fdd files
376 or whatever...
377
378 =====
379 #!/bin/sh
380
381 rm -f source2e.gls source2e.ind source2e.toc
382
383 # First run:
384 # Create new standard ltxdoc.cfg file
385 # Pass the (possibly empty) list of arguments supplied on the
386 # command line to article class.
387 #
388 # If you use A4 paper, running this script with argument
389 # a4paper
390 # may save about 30 pages.
391 #
392 echo "\PassOptionsToClass{${*}}{article}" > ltxdoc.cfg
393
394
395 # Now LaTeX the file with this cfg file.
396 #
397 latex source2e.tex
398
399
400 # Make the Change log and Glossary.
401 #

```

```

402 makeindex -s source2e.ist source2e.idx
403 makeindex -s gglo.ist -o source2e.gls source2e.glo
404
405
406 # Second run: append \includeonly{} to ltxdoc.cfg to speed up things
407 # (this run needed only to get changes and index listed in .toc file)
408 #
409 # Note that the index will not be made incorrect by the insertion
410 # of the table of contents as the front matter uses a diferent page
411 # numbering scheme.
412 #
413 echo "\includeonly{}" >> ltxdoc.cfg
414
415 latex source2e.tex
416
417
418 # Third and final run, to put everything together.
419 # First restore the cfg file:
420 #
421 echo "\PassOptionsToClass{${*}}{article}" > ltxdoc.cfg
422 latex source2e.tex
423
424
425 =====
426 #!/bin/sh
427
428 # Running this script will process all the dtx fdd and *guide.tex
429 # and ltnews*.tex files in the LaTeX distribution, except the dtx
430 # files included in source2e.tex.
431 # (The shell first script in the comments of source2e.tex will
432 # process those.)
433
434 # Any command line arguments (eg a4paper) are taken as options to the
435 # article class.
436
437 # This script is likely to take ages!
438
439 echo "\PassOptionsToClass{${*}}{article}" > ltxdoc.cfg
440 echo "\batchmode" >> ltxdoc.cfg
441
442 # The next four lines produce full indexes and change logs
443 # you may not want those.
444 echo "\AtBeginDocument{\RecordChanges}" >> ltxdoc.cfg
445 echo "\AtEndDocument{\PrintChanges}" >> ltxdoc.cfg
446 echo "\AtBeginDocument{\CodelineIndex\EnableCrossrefs}" >> ltxdoc.cfg
447 echo "\AtEndDocument{\PrintIndex}" >> ltxdoc.cfg
448
449 # If you do not want any code listings, just documentation, then instead
450 # of the above four lines, uncomment the following:
451 # echo "\AtBeginDocument{\OnlyDescription}" >> ltxdoc.cfg
452
453 echo "\PassOptionsToClass{${*}}{article}" > ltxguide.cfg
454 echo "\batchmode" >> ltxguide.cfg
455
456 cp ltxguide.cfg ltnews.cfg
457
458
459 for i in *dtx *fdd *guide.tex ltnews*.tex
460 do
461 B='basename $i .dtx'
462
463 if (grep "Include{${B}}" source2e.tex >/dev/null ; )
464 then
465 echo In source2e: $i
466 else
467 echo latex $i
468 if (latex $i > /dev/null)
469 then
470 echo latex $i
471 latex $i > /dev/null
472 echo makeindex -s gind.ist ${B}.idx
473 makeindex -s gind.ist ${B}.idx > /dev/null 2> /dev/null
474 echo makeindex -s gglo.ist -o ${B}.gls ${B}.glo
475 makeindex -s gglo.ist -o ${B}.gls ${B}.glo > /dev/null 2> /dev/null

```

```
476     echo latex $i
477     latex $i > /dev/null
478     else
479     echo "!!! LaTeX ERROR: $i. (See $B.log.)"
480     fi
481 fi
482
483 done
```

classes.dtx

ランポートさんの本には「source2e.texにも、classes.dtxのことが書かれている」とありますが、(少なくとも現時点の)source2e.texには特には書いてないみたいですね。

いずれにしても、“kernel”には定義されていない(または、kernelではその準備だけがされている)各種クラスファイルに依存するいろいろな設定の説明については、“classes.dtx”のほうに書いてあります。

“classes.dtx”は、article.cls、report.cls、book.cls、size10.clo、size11.clo、size12.clo、bk10.clo、bk11.clo、bk12.cloの、ソースです。

第4章 pL^AT_EX 2_ε

Ken Nakano 「pL^AT_EX 2_εについて」 (作成日: 1997/01/29) [platex.dtx]

第1節 概要

この文書は、pL^AT_EX 2_εの概要を示していますが、使い方のガイドではありません。pL^AT_EX 2_εの機能についての説明は、中野賢『日本語 L^AT_EX 2_εブック』(アスキー・1996年)を参照してください。日本語 T_EX については、アスキー出版技術部責任編集『日本語 T_EX テクニカルブック I』(アスキー・1990年)を参照してください。

pL^AT_EX 2_εでは、インプレス・ラボ監修・アスキー書籍編集部編『縦組対応パーソナル日本語 T_EX』(アスキー出版局・1994年)で説明されている、いくつかの拡張コマンドの動作を修正しています。その詳細については、pl_{ext}.dtx を参照してください。

L^AT_EX の機能については、Laslie Lamport, “L^AT_EX: A Document Preparation System”, Addison-Wesley, second edition, 1994 や Michel Goossens, Frank Mittelbach, Alexander Samarin, “The L^AT_EX Companion”, Addison-Wesley, 1994 などを参照してください。新しい機能については usrguide.tex を参照してください。

この文書の構成は次のようになっています。

第1節 この節です。この文書についての概要と、DOCSTRIP のためのオプションについて述べています。

第2節 pL^AT_EX 2_εで拡張した機能についての概要です。付属のクラスファイルやパッケージファイルについても簡単に説明しています。

第3節 旧バージョンの pL^AT_EX との互換性について述べています。

付録 A pL^AT_EX 2_εの dtx ファイルをまとめて一つの DVI ファイルにするための文書ファイル説明をしています。

付録 B 付録 A で説明をした文書ファイル进行处理する sh スクリプト(手順)、DOCSTRIP 文書ファイル内の入れ子の対応を調べる perl スクリプトなどについて説明しています。

1.1 DOCSTRIP プログラムのためのオプション

この文書を DOCSTRIP プログラムによって処理することによって、いくつかの異なるファイルを生成することができます。

この文書の DOCSTRIP プログラムのためのオプションは、次のとおりです。

オプション	意味
plcore	フォーマットファイルを作るためのファイルを生成
pldoc	pL ^A T _E X 2 _ε のソースファイルをまとめて組版するための文書ファイルを生成
shprog	上記のファイルを作成するための sh スクリプトを生成
plprog	入れ子構造を調べる簡単な perl スクリプトを生成
Xins	上記の sh スクリプトや perl スクリプトを取り出すための DOCSTRIP バッチファイルを生成

1.1.1 ファイルの取り出し方

たとえば、この文書の “plcore” の部分を “platex.ltx” というファイルにするときの手順はつぎのようになります。

1. platex docstrip
2. 入力ファイルの拡張子 (dtx) を入力する。
3. 出力ファイルの拡張子 (ltx) を入力する。
4. DOCSTRIP オプション (plcore) を入力する。
5. 入力ファイル名 (platex) を入力する。
6. platex.ltx が存在する場合は、確認を求めてくるので、“y” を入力する。
7. 別の処理を行なうかを問われるので、“n” を入力する。

これで、platex.ltx が作られます。

あるいは、次のような内容のファイル batch.ins を作成し、platex fmt.ins することでも platex.ltx を作成することができます。

```
\def\batchfile{batch.ins}
\input docstrip.tex
\generateFile{platex.ltx}{t}{\from{platex.dtx}{plcore}}
```

DOCSTRIP プログラムの詳細は、docstrip.dtx を参照してください。

第2節 pL^AT_EX 2_εの機能について

pL^AT_EX 2_εの機能は、いくつかのファイルに分割されて実装されています。これらのファイルはつぎの3種類に分類することができます。

- フォーマットファイル
- クラスファイル
- パッケージファイル

フォーマットファイルには、基本的な機能が定義されており、p^LA^TE_X 2_εの核となるファイルです。このファイルに定義されているマクロは、実行時の速度を高めるために、あらかじめ T_EX の内部形式の形で保存されています。

クラスファイルとパッケージファイルは、従来、スタイルファイルと呼ばれていたものです。L^AT_EX 2_εではそれらを、レイアウトに関するものをクラスファイルと呼び、マクロの拡張をするものをパッケージファイルと呼んで区別するようになりました。

T_EX 文書が使用するクラスは、文書のプリアンブルで `\documentclass` コマンドを用いて指定します。`\documentclass` ではなく、旧版の `\documentstyle` を用いると、自動的に 2.09 互換モードに入ります。互換モードは旧版の文書を組版するためだけに作られていますので、新しく文書を作成する場合は、`\documentclass` コマンドを用いてください。互換モードでは L^AT_EX の新機能も使えなくなります。

旧版では `\documentstyle` のオプションでマクロファイルを読み込んでいましたが、L^AT_EX では、`\usepackage` コマンドを用いて読み込みます。

2.1 フォーマットファイル

フォーマットファイルには、基本的な機能が定義されていますが、これらは T_EX の内部形式に変換された形式となっています。フォーマットファイルを作成するには、ソースファイル “`platex.ltx`” を `inipltex` プログラムで処理します。

次のリストが、その内容です。ただし、このバージョンでは、L^AT_EX から p^LA^TE_X 2_ε への拡張を `plcore.ltx` をロードすることで行ない、`latex.ltx` には直接、手を加えないようにしています。したがって `platex.ltx` はとても短いものとなっています。`latex.ltx` には L^AT_EX のコマンドが、`plcore.ltx` には p^LA^TE_X 2_ε で拡張したコマンドが定義されています。

```

1 <*plcore>
2 \let\orgdump\dump
3 \let\dump\relax
4 \input latex.ltx
5 \typeout
   {*****~J%
6     *~J%
7     * making pLaTeX format~J%
8     *~J%
9     *****}
10 \makeatletter
11 \input plcore.ltx
12 \makeatother
13 \the\everyjob
14 \let\dump\orgdump
15 \dump
16 <plcore>\endinput
17 </plcore>
```

実際に p^LA^TE_X 2_εへの拡張を行なっている `plcore.ltx` は、DOCSTRIP プログラムによって、次のファイルの断片が連結されたものです。

`plvers.dtx` は、p^LA^TE_X 2_εのフォーマットバージョンを定義しています。

`plfonts.dtx` は、NFSS2 を拡張しています。

`plcore.dtx` は、上記以外のコマンドでフォーマットファイルに格納されるコマンドを定義しています。

プリロードフォントや組版パラメータなどの設定は、`pldefs.ltx` をロードすることで行なっています。このファイルに記述されている設定を変更すれば、p^LA^TE_X 2_ε をカスタマイズすることができます。カスタマイズする場合は、このファイルを直接、修正するのではなく、`pldefs.cfg` という名前でコピーをして、そのファイルを編集します。`pldefs.cfg` は `pldefs.ltx` の代わりに読み込まれます。

2.1.1 バージョン

p^LA^TE_X 2_εのバージョンやフォーマットファイル名は、`plvers.dtx` で定義しています。

2.1.2 NFSS2 コマンド

L^AT_EX では、フォント選択機構として NFSS2 を用いています。p^LA^TE_X 2_εでは、オリジナルの NFSS2 と同様のインターフェイスで、和文フォントを選択できるように、`plfonts.dtx` で NFSS2 を拡張しています。

p^LA^TE_X 2_εの NFSS2 は、フォントを切替えるコマンドを指定するときに、それが欧文書体か和文書体のいずれかを対象とするものかを、できるだけ意識しないようにする方向で拡張しています。いいかえれば、コマンドが(可能な限りの)判断をします。したがって数多くある英語版のクラスファイルやパッケージファイルなどで書体の変更を行っている箇所を修正する必要はあまりありません。

`plfonts.dtx` ファイルでは、NFSS2 コマンドの定義のほか、プリロードフォントの設定、和文エンコードの定義、組版パラメータなどの設定、フォント定義ファイルなどの記述も含まれています。

NFSS2 についての詳細は、L^AT_EX 2_εに付属の `fntguide.tex` を参照してください。

2.1.3 出カルーチンとフロート

`plcore.dtx` は、次の項目に関するコマンドを日本語処理用に修正や拡張をしています。

プリアンブルコマンド
改ページ
改行
オブジェクトの出力順序
トンボ
脚注マクロ
相互参照
疑似タイプ入力

2.2 クラスファイルとパッケージファイル

クラスファイルとパッケージファイルは、従来、スタイルファイルと呼ばれていたものです。L^AT_EX ではそれらを、レイアウトに関するものをクラスファイルと呼び、マクロの拡張をするものをパッケージファイルと呼んで区別するようになりました。

p^LA^TE_X 2_εが提供する、クラスファイルやパッケージファイルのいくつかは、オリジナルのファイルを修正しています。修正箇所には “`platex`” 条件が付けられています。

p^LA^TE_X 2_εに付属のクラスファイルは、次のとおりです。

`jbook.cls`, `jarticle.cls`, `jreport.cls`
横組用の標準クラスファイル。`jclasses.dtx` から作成される。

tbook.cls, tarticle.cls, treport.cls
縦組用の標準クラスファイル。jclasses.dtx から作成される。

jltxdoc.cls
.dtx ファイルを組版するためのクラスファイル。
jltxdoc.dtx から作成される。

jltxguid.cls
usrguide.tex や fntguide.tex などを組版するためのクラスファイル。

また、p \LaTeX 2 ϵ に付属のパッケージファイルは、次のとおりです。

oldpfont.sty
p \LaTeX 2.09 のフォントコマンドを提供するパッケージ。oldpfont.dtx から作成される。

ptrace.sty
tracefnt.sty で再定義された NFSS2 コマンドを p \LaTeX 2 ϵ 用に再々定義するためのパッケージ。

ascmac.sty, tascmac.sty
旧バージョンの p \LaTeX で配布されていたファイル。

pnext.sty
縦組用の拡張コマンドなどが定義されているファイル。

第 3 節 旧バージョンとの互換性

ここでは、このバージョンと以前のバージョンとの互換性や拡張部分について説明をしています。

3.1 p \LaTeX 2.09 との互換性

p \LaTeX 2 ϵ は、 \LaTeX の上位互換という形を取っていますが、いくつかのパラメータなども変更しています。したがって英文書など、 \LaTeX でも処理できるファイルを p \LaTeX 2 ϵ で処理しても、完全に同じ結果になるとは限りません。これは、英語版の \LaTeX でも同じです。詳細は、 \LaTeX 2 ϵ に付属の usrguide.tex を参照してください。

多くのクラスファイルやパッケージファイルはそのまま使えると思います。ただし、それらが p \LaTeX 2 ϵ で拡張しているコマンドと同じ名前のコマンドを再定義している場合は、コマンドの拡張の仕方によってはエラーになることもあります。用いようとしている、クラスファイルやパッケージファイルがうまく動くかどうかを、完全に確かめる方法は残念ながらありません。一番簡単なのは、動かしてみることです。不幸にもうまく動かない場合は、ログファイルや付属の文書ファイルを参考に原因を調べてください。

付録 A 文書ファイル

ここでは、このパッケージに含まれている dtx ファイルをまとめて組版するための文書ファイルについて説明をしています。個別に処理した場合と異なり、変更履歴や索引も付きます。全体で、およそ 150 ページ程度になります。

[...以下略。「付録 A」は “pldoc.tex” の解説です。続きはお手許のファイルをご覧ください。pldoc.tex 自体は、次頁以降に挙げてあります。]

付録 B おまけプログラム

[略。お手許のファイルをご覧ください。]

この “platex.dtx” というファイルは、p \LaTeX の開発者である中野賢さんご自身が書かれた p \LaTeX を概説するドキュメントなので、p \LaTeX を使われる方はまず最初に読むべきものだと思うのですが、必ずしもそうはなっていないみたいですので、ここに再録させていただきました（このファイルも、どなたの PC の中にもあると思います。なお、オリジナルの文書の見出しはただ「1, 2, 3, A, B」となっているのですが、ここでは勝手に「第 1 節、第 2 節、第 3 節、付録 A、付録 B」とさせていただいています。申し訳ありません）。

次頁以降には、“pldoc.tex” を listings に流し込んだものを掲載しています。上の「付録 A」の部分に書かれていますように、pldoc.tex を p \LaTeX で処理すると、全体で 150 ページほどの “The p \LaTeX 2 ϵ Sources” というドキュメントになります（目次や変更履歴、コマンドインデックスも付いています）。

“source2e.tex” と “classes.dtx” は別ファイルですが、こちらの “pldoc.tex” のほうには、“jclasses.dtx” も含まれます（154 行目に \DocInclude{jclasses} とありますよね。もちろん、jclasses.dtx だけを別個にタイプセットすることも出来ます）。

source2e.tex のところにも書きましたが、hypdoc パッケージを使ってタイプセットすると、リンク付きのドキュメントになりますので、検索にとっても便利になります。

pldoc.tex

```

1  %%
2  %% This is file 'pldoc.tex',
3  %% generated with the docstrip utility.
4  %%
5  %% The original source files were:
6  %%
7  %% platex.dtx (with options: 'pldoc')
8  %%
9  %% IMPORTANT NOTICE:
10 %%
11 %% For the copyright see the source file.
12 %%
13 %% Any modified versions of this file must be renamed
14 %% with new filenames distinct from pldoc.tex.
15 %%
16 %% For distribution of the original source see the terms
17 %% for copying and modification in the file platex.dtx.
18 %%
19 %% This generated file may be distributed as long as the
20 %% original source files, as listed above, are part of the
21 %% same distribution. (The sources need not necessarily be
22 %% in the same archive or directory.)
23 %% File: platex.dtx
24 %% \CharacterTable
25 %% {Upper-case  A\B\C\D\E\F\G\H\I\J\K\L\M\N\O\P\Q\R\S\T\U\V\W\X\Y\Z
26 %% Lower-case  a\b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
27 %% Digits      0|1|2|3|4|5|6|7|8|9
28 %% Exclamation  !      Double quote  "      Hash (number)  \#
29 %% Dollar       \$      Percent      \%      Ampersand     \&
30 %% Acute accent \'     Left paren   \langle   Right paren   \rangle
31 %% Asterisk     *      Plus         \+       Comma         \,
32 %% Minus        -      Point       \.       Solidus       \/.
33 %% Colon        :      Semicolon   \;       Less than     \<
34 %% Equals       =      Greater than \>      Question mark \?
35 %% Commercial at \@    Left bracket \[       Backslash     \|
36 %% Right bracket \]    Circumflex  \^       Underscore    \_
37 %% Grave accent  `     Left brace  \{       Vertical bar  \|
38 %% Right brace   }     Tilde       \~}
39 %%
40 \begin{filecontents}{pldoc.dic}
41 西暦   せいれき
42 和暦   われき
43 \end{filecontents}
44 \documentclass{jltxdoc}
45 \usepackage{plex}
46 \listfiles
47
48 \DoNotIndex{\def,\long,\edef,\xdef,\gdef,\let,\global}
49 \DoNotIndex{\if,\ifnum,\ifdim,\ifcat,\ifmmode,\ifvmode,\ifhmode,%
50             \iftrue,\iffalse,\ifvoid,\ifx,\ifeof,\ifcase,\else,\or,\fi}
51 \DoNotIndex{\box,\copy,\setbox,\unvbox,\unhbox,\hbox,%
52             \vbox,\vtop,\vcenter}
53 \DoNotIndex{\@empty,\immediate,\write}
54 \DoNotIndex{\egroup,\bgroup,\expandafter,\begingroup,\endgroup}
55 \DoNotIndex{\divide,\advance,\multiply,\count,\dimen}
56 \DoNotIndex{\relax,\space,\string}
57 \DoNotIndex{\csname,\endcsname,\@spaces,\openin,\openout,%
58             \closein,\closeout}
59 \DoNotIndex{\catcode,\endinput}
60 \DoNotIndex{\jobname,\message,\read,\the,\m@ne,\noexpand}
61 \DoNotIndex{\hsize,\vsize,\hskip,\vskip,\kern,\hfil,\hfill,\hss,\vss,\unskip}
62 \DoNotIndex{\m@ne,\z@,\z@skip,\@ne,\tw@,\p@,\@minus,\@plus}
63 \DoNotIndex{\dp,\wd,\ht,\setlength,\addtolength}
64 \DoNotIndex{\newcommand,\renewcommand}
65
66 \IndexPrologue{\part*{索引}%
67               \markboth{索引}{索引}%
68               \addcontentsline{toc}{part}{索引}%
69 イタリアック体の数字は、その項目が説明されているページを示しています。
70 下線の引かれた数字は、定義されているページを示しています。

```

```

71 その他の数字は、その項目が使われているページを示しています。}
72 \GlossaryPrologue{\part*{変更履歴}%
73     \markboth{変更履歴}{変更履歴}%
74     \addcontentsline{toc}{part}{変更履歴}}
75
76 \makeatletter
77 \def\changes@#1#2#3{%
78     \let\protect\@unexpandable@protect
79     \edef\@tempa{\noexpand\glossary{#2\space\currentfile\space#1\levelchar
80         \ifx\saved@macroname\@empty
81             \space\actualchar\generalname
82         \else
83             \expandafter\@gobble
84             \saved@macroname\actualchar
85             \string\verb\quotechar*%
86             \verbatimchar\saved@macroname
87             \verbatimchar
88         \fi
89         :\levelchar #3}}%
90     \@tempa\endgroup\@esphack}
91 \makeatother
92 \RecordChanges
93 \CodelineIndex
94 \EnableCrossrefs
95 \setcounter{IndexColumns}{2}
96 \settowidth\MacroIndent{\ttfamily\scriptsize 000\ }
97 \begin{document}
98 \title{The p\LaTeX\ Sources}
99 \author{Ken Nakano}
100
101 \newcommand{\includeltpatch}{%
102     \def\currentfile{plpatch.ltx}
103     \part{plpatch}
104     {\let\ttfamily\relax
105     \xdef\filekey{\filekey, \thepart={\ttfamily\currentfile}}}%
106     Things we did wrong\ldots
107     \IndexInput{plpatch.ltx}}
108
109 \makeatletter
110 \let\patchdate=\@empty
111 \begingroup
112     \def\ProvidesFile#1\pfmtversion#2{\date{#2}\endinput}
113     \input{plvers.dtx}
114 \global\let\X@date=\@date
115
116     \long\def\Xdef#1#2#3\def#4#5{%
117     \xdef\X@date{#2}%
118     \xdef\patchdate{#5}%
119     \endinput}%
120     \InputIfFileExists{plpatch.ltx}
121     {\let\def\Xdef}{\global\let\includeltpatch\relax}
122 \endgroup
123
124 \ifx\@date\X@date
125     \def\Xpatch{0}
126     \ifx\patchdate\Xpatch\else
127         \edef\@date{\@date\space Patch level\patchdate}
128     \fi
129 \else
130     \@warning{plpatch.ltx does not match plvers.dtx!}
131     \let\includeltpatch\relax
132 \fi
133 \makeatother
134
135 \pagenumbering{roman}
136 \maketitle
137 \renewcommand\maketitle{}
138 \tableofcontents
139 \clearpage
140 \pagenumbering{arabic}
141
142 \DocInclude{plvers} % pLaTeX version
143
144 \DocInclude{plfonts} % NFSS2 commands

```

```
145
146 \DocInclude{plcore} % kernel commands
147
148 \DocInclude{plxt} % external commands
149
150 \DocInclude{pl209} % 2.09 compatibility mode commands
151
152 \DocInclude{kinsoku} % kinsoku parameter
153
154 \DocInclude{jclasses} % Standard class
155
156 \DocInclude{jltxdoc} % dtx documents class
157
158 \includeltpatch % patch file
159
160 \StopEventually{\end{document}}
161
162 \clearpage
163 \pagestyle{headings}
164 \hbadness=10000
165 \newcount\hbadness
166 \hfuzz=\maxdimen
167 \PrintChanges
168 \clearpage
169 \begingroup
170   \def\endash{--}
171   \catcode'\-\active
172   \def-{\futurelet\temp\indexdash}
173   \def\indexdash{\ifx\temp-\endash\fi}
174
175   \PrintIndex
176 \endgroup
177 \let\PrintChanges\relax
178 \let\PrintIndex\relax
179 \end{document}
180 \endinput
181 %%
182 %% End of file 'pldoc.tex'.
```