This is a list of all substantial corrections made to *Computers & Typesetting* since the beginning of 2014. (More precisely, it lists errors corrected since the 19th printing of Volume A, the 9th printing of Volume B, the 8th printing of Volume C, the 6th printing of Volume D, and the 7th printing of Volume E. But it omits changes that are “purely cosmetic.”) Corrections made to the softcover version of *The \TeX\book*, beginning with its 32nd printing, are the same as corrections to Volume A. Corrections to the softcover version of *The \METAFONT\book*, beginning with its 11th printing, are the same as corrections to Volume C. Changes to the mini-indexes and master indexes of Volumes B, D, and E are not shown here unless they are not obviously derivable from what has been shown. Some (or all) of these errors have been corrected in the most recent printings.

---

Page A34, line 3 from the bottom (01/09/20)

not, you can say \texttt{\errorcontextlines=100 \oops} and try again. (That will usually

Page A43, line 6 (07/24/14)

keyboard, or that have been preempted for formatting?

Page A49, cummings quote (08/03/19)

(delete the period at the end of the line)

Page A66, line 3 from the bottom (08/26/17)

Such displays of box contents will be discussed further in Chapters 12 and 27.

Page A105, lines 9–16 (01/16/21)

\textbf{\textcolor{red}{WARNING}} If you say \texttt{\vadjust{⟨vertical mode material⟩}} within a paragraph, \TeX will use internal vertical mode to insert the specified material into the vertical list that encloses the paragraph, immediately after whatever line contained the position of the \texttt{\vadjust}. For example, you can say ‘\texttt{\vadjust{⟨kern1pt⟩}’ to increase the amount of space between lines of a paragraph if those lines would otherwise come out too close together. (The author did that in the current line, just to illustrate what happens.) Also, if you want to make sure that a page break will occur immediately after a certain line, you can say ‘\texttt{\vadjust{\eject}’ anywhere in that line.

Page A122, lines 3–8 (11/24/19)

\texttt{\count255, \dimen255, \skip255, \muskip255, and \toks255} are traditionally kept available for such purposes. Furthermore, plain \TeX reserves \texttt{\dimen0} to \texttt{\dimen9}, \texttt{\skip0} to \texttt{\skip9}, \texttt{\muskip0} to \texttt{\muskip9}, and \texttt{\box0} to \texttt{\box9} for “scratchwork”; these registers are never allocated by the \texttt{\new...} operations. We have seen that \texttt{\count0} through \texttt{\count9} are special, and \texttt{\box0} also turns out to be special; so those registers should be avoided unless you know what you are doing.
\[\mathopen{\hbox{$\left#1\langle$\strut$\right.$}}\]

Page A155, line 8 from the bottom  (01/17/21)

Page A155, the bottom six lines  (12/10/18)

Individual symbols; \texttt{\textbackslash left…\textbackslash right} constructions are treated as “inner” subformulas, which means that they will be surrounded by additional space in certain circumstances. All other subformulas are generally treated as ordinary symbols, whether they are formed by \texttt{\overline} or \texttt{\hbox} or \texttt{\vcenter} or by simply being enclosed in braces. Thus, \texttt{\mathord} isn’t really a necessary part of the \TeX{} language; instead of typing ‘$1\mathord,234$’ you can get the same effect from ‘$1{},234$’.

Page A155, the bottom six lines  (12/10/18)

Page A158, line 19  (12/10/18)

\texttt{Inner} is an inner atom produced by ‘\texttt{\textbackslash left…\textbackslash right};

Page A170, lines 18 and 19  (12/10/18)

Subformulas delimited by \texttt{\textbackslash left} and \texttt{\textbackslash right} are treated as type Inner. The following table is used to determine the spacing between pairs of adjacent atoms:

Page A171, line 19 from the bottom  (06/15/19)

Formula produces a result essentially equivalent to ‘\texttt{\textbackslash left((subformula)\textbackslash right)}’, when

Page A215, line 16 from the bottom becomes two lines  (10/13/20)

\begin{itemize}
  \item Just after a token such as \texttt{\$3} that begins math mode, to see if another token of category 3 follows.
\end{itemize}

Page A222, lines 21–23  (01/16/21)

\begin{itemize}
  \item \texttt{\hbox{\langle box specification\rangle\{}\langle horizontal mode material\rangle\}} \quad \text{(see Chapter 12)}
  \item \texttt{\vbox{\langle box specification\rangle\{}\langle vertical mode material\rangle\}} \quad \text{(see Chapter 12)}
  \item \texttt{\vtop{\langle box specification\rangle\{}\langle vertical mode material\rangle\}} \quad \text{(see Chapter 12)}
\end{itemize}

Page A222, lines 11–13 from the bottom  (01/16/21)

Type \texttt{\split} operation is also explained in Chapter 15. In math modes an additional type of box is available: \texttt{\vcenter{\langle box specification\rangle\{}\langle vertical mode material\rangle\}} \quad \text{(see Chapter 17)}.

Page A222, lines 21–23  (01/16/21)

\begin{itemize}
  \item \texttt{\hbox\langle box specification\rangle\{}\langle horizontal mode material\rangle\}} \quad \text{(see Chapter 12)}
  \item \texttt{\hbox\langle box specification\rangle\{}\langle vertical mode material\rangle\}} \quad \text{(see Chapter 12)}
  \item \texttt{\vbox\langle box specification\rangle\{}\langle vertical mode material\rangle\}} \quad \text{(see Chapter 12)}
\end{itemize}

Page A222, lines 11–13 from the bottom  (01/16/21)

Page A232, line 14  (01/10/21)

Tabs outside; \texttt{\textbackslash global\settabs} ‘will not do what you might think it should.

Page A232, line 14  (01/10/21)

\begin{itemize}
  \item Tabs outside; \texttt{\textbackslash global\settabs} ‘will not do what you might think it should.
\end{itemize}

Page A233, lines 3–5  (04/27/15)

Only two tabs are set in this case, because only two \texttt{&}'s appear in the sample line. (A sample line usually ends with \texttt{\&\cr}, as it does here, because text material between the last tab and \texttt{\cr} isn’t used for anything.)
Page A252, lines 5–7 (12/25/20)
blank, and the footline is normally a centered page number, but you can specify any headline and footline that you want by changing the token lists \headline and \footline. For example,

Page A253, lines 7–9 from the bottom (10/27/20)

\everypar or \errhelp, except that TeX retains the begin-group symbol ‘{’ at the beginning and the end-group symbol ‘}’ at the end. These grouping characters help to keep the output routine from interfering with what TeX was doing.

Page A256, line 19 (08/28/15)

\baselineskip=24pt \lineskiplimit=0pt

Page A277, lines 9 and 10 from the bottom (08/26/17)

⟨hyphenation assignment⟩−→ \hyphenation(filler)\{\hyphenations\}
| \patterns(filler)\{\patterns\}

Page A286, bottom two lines (and affecting the top lines of page 287) (08/26/17)

stands for zero or more ⟨assignment⟩ commands other than \setbox, possibly with ⟨filler⟩. If the assignments are not followed by a ⟨character⟩, where ⟨character⟩ stands

Page A287, lines 11–17 (04/22/20)

\discretionary{⟨disc text⟩}{⟨disc text⟩}{⟨disc text⟩}. This command has the same effect as in horizontal mode (see Chapter 25), but the third ⟨disc text⟩ must produce an empty list.

Page A290, line 11 from the bottom (11/01/20)
is corrupted or was prepared for a different version of TeX.

Page A305, bottom line (06/30/20)

\setbox0=\hbox{#1}\advance\dimen0 by \wd0 \).
represent text entered from the user’s terminal, or with ‘<insert>’, when they represent text inserted during error recovery).

(The next line must also not be too tall.) Here \specialstar is a box of height zero and depth \strutdepth, and it puts an asterisk in the left margin:

17.21. Assigning \delcode’{ would not work to allow ‘\left{’, because the brace has category 1 and isn’t a legal ⟨delim⟩. Allowing brace delimiters would be a bad idea because it would mess up other constructions, such as arguments to macros, and components of alignments. Moreover, a user who gets away with ‘\left{’ is likely to try also ‘\bigl{’, which fails miserably.

its natural width. The \hbox version also invokes \everyhbox and \everymath.

the three tokens !1, #2, [1; the (replacement text) consists of the six tokens #1, #6,

is otherwise irrelevant. Thus, ‘\def\!!1#2[#1[#2]\!!2]’ would produce an essentially

final parameter in the parameter text; ‘!1’ would have been rendered ‘#1’.

21.10. If you say ‘\let\the=0\edef\next\write\cont{⟨token list⟩}\next’, the \write will be executed after \edef expands everything except \the.

\+&{\bf end};\cr % note that the semicolon isn’t bold

of plain \TeX format; but some of them are primitive (built in), such as ‘\par’ (end of paragraph), ‘\noindent’ (beginning of non-indented paragraph), and ‘/’ (italic
Braces are used for grouping, when supplying arguments to macros; so they cannot also be used as math delimiters, or as arguments to macros such as $\texttt{\textbackslash big}$. (One could change their catcodes to 12, and use some other pair of characters for grouping; but that would not be plain \TeX.)

(2) The registers $\texttt{\textbackslash count255, \textbackslash dimen255, \textbackslash skip255, \textbackslash toks255}$, and $\texttt{\textbackslash muskip255}$ are freely available in the same way. (3) All assignments to the scratch registers whose numbers are 1, 3, 5, 7, and 9 should be $\texttt{\textbackslash global}$; all assignments to the other scratch registers (0, 2, 4, 6, 8, 255) should be non-$\texttt{\textbackslash global}$. (This prevents the phenomenon of “save stack buildup” discussed in Chapter 27.) (4) Furthermore, it’s possible to use any register in a group, if you ensure that \TeX’s grouping mechanism will restore the register when you’re done with the group, and if you are certain that other macros will not make global assignments to that register when you need it. (5) But when a register is used by several macros, or over long spans of time, it should be allocated by $\texttt{\textbackslash newcount, \textbackslash newdimen, \textbackslash newbox}$, etc. (6) Similar remarks apply to input/output streams used by $\texttt{\textbackslash read}$ and $\texttt{\textbackslash write}$, to math families used by $\texttt{\textbackslash fam}$, to sets of hyphenation rules used by $\texttt{\textbackslash language}$, and to insertions (which require $\texttt{\textbackslash box, \textbackslash count, \textbackslash dimen}$, and $\texttt{\textbackslash skip}$ registers all having the same number).

\def\wlog{\immediate\write-1 } % this will write on log file (only)
\outer\def\newmuskip{\alloc@3\muskip\muskipdef\@cclv}
\outer\def\newtoks{\alloc@5\toks\toksdef\@cclv}
\def\fmtversion{3.1415926535} % identifies the current format

close as possible to the ASCII conventions. (b) Make sure that codes ‘041–046, ‘060–071, ‘136, ‘141–146, and ‘160–171 are present and that each unrepresentable in-
and \if...\fi tests, as well as special operations like \the and \input, while the latter category includes the primitive commands listed in Chapters 24–26. The expansion of

\$\text{general display}\$ to be invoked, with \eq defined to be \(\alpha\). Furthermore, when an equation number \(\beta\) is present, it should be stored in \eqn, and the test \ifeqno should be true. In such cases \ifeqno should distinguish \leqno from \eqno. Here

\setbox2=\lastbox \setbox\footins=\vbox{\box2}

since \lastbox will be the result of \rigidbalance, which is an hbox.

The computer file \texttt{texbook.tex} that generated \textit{The \TeXbook} begins with a

\TeX commands that look like this in the file \texttt{texbook.tex}:

\def\bull{\vrule height.9ex width.8ex depth-.1ex \relax} % square bullet

\vrule height6pt depth2pt width0pt \relax} % a strut for \insert\margin

15e. Enclose the vbox that was constructed in Rule 15c or 15d by delimiters (\(\lambda, \rho\)) whose height plus depth is at least \(\sigma_20\), if \(C > T\), and at least \(\sigma_21\) otherwise. Shift the delimiters up or down so that they are vertically centered with respect to the axis. Replace the generalized fraction by an Ord atom whose nucleus is the resulting sequence of three boxes (\(\lambda, \vbox, \rho\)). Go to rule 19.

atom and the right boundary item to a Close atom. The entire resulting list now becomes the nucleus of an Inner atom. (All of the calculations in this step are done with \(C\) equal to the starting style of the math list; style items in the middle of the list do not affect the style of the right boundary item.)
of the process; the trial word consists of all the letters found in admissible items, up to a maximum of 63. Notice that all of these letters are in font \( f \).

Page A458 and following, selected amendments to the index

\[1\] (progress report), 23, 119.
\( \text{\aa} \) (˚a), 52, 356.
\( \text{\AA} \) (˚A), 52, 356.
(disc text), 287, 292.
(horizontal mode material), 278, 285, 287.
integral signs, see \\int, \\oint, \\smallint.
(math mode material), 287, 289–293.
\( \text{\ae} \), 311, 312, 316, 332, 335, 351, 354, 360–362, 419.
\( \text{\ø} \), 52, 356.
\( \text{\O} \) (Ø), 52, 356.
programs, for computers, 38, 165, 234.
repeating templates, see periodic preambles.
replacement text, 200–204, 212, 280, 300, 329.
right delimiters, see closings.
struts, 82, 125, 131, 142, 155, 178, 245–247, 255, 329, 416, 422, 423.
(vertical mode material), 278, 280–282, 290.

Page Bv (formerly Bvi), bottom two lines

all of those changes. I now believe that the final bug was discovered on 22 October 2020 and removed in version 3.141592653. The finder’s fee has converged to $327.68.

Page B2, line 10 from the bottom

\define \textit{banner} \texttt{"This is \TeX, Version 3.141592653"} \{ printed when \TeX\ starts \}

Page B4, line 8 of §7

diagnostic information for \texttt{\tracingparagraphs}, \texttt{\tracingpages}, and \texttt{\tracingrestores}.

Page B21, lines 33 and 34

\texttt{[‘41 \rightarrow ‘46, ‘60 \rightarrow ‘71, ‘136, ‘141 \rightarrow ‘146, ‘160 \rightarrow ‘171]} must be printable. Thus, at least 80 printable characters are needed.

Page B28, lines 3 and 4

not serious since we assume that this part of the program is system dependent.

Page B28, line 2 from the bottom

\texttt{\textbf{var} \( k \): 0 .. 23; \{ index to current digit; we assume that \( |n| < 10^{23} \} \)
Page B35, line 2 of §83 becomes two lines (06/27/20)

\hspace{1cm} \textbf{loop begin continue: if interaction \neq error_stop_mode then return;}
\hspace{1cm} \textit{clear_for_error_prompt; prompt_input(“?”);} 

Page B36, line 11 of §84 (07/03/20)

\hspace{1cm} “E": if base_ptr > 0 then if input_stack[base_ptr].name_field \geq 256 then 

Page B36, line 5 of §85 becomes two lines (07/03/20)

\hspace{1cm} if base_ptr > 0 then if input_stack[base_ptr].name_field \geq 256 then print("E to edit your file.") 

Page B40, line 5 from the bottom (08/07/20)

\hspace{1cm} \langle \text{Try to insert an instruction for me, e.g., I\showlists}, \rangle 

Page B58, lines 2 and 3 of §136 (10/11/20)

\hspace{1cm} the values corresponding to ‘\hbox{}’. The \texttt{sub_type} field is set to \texttt{minQuarterword}, for historic reasons that are no longer relevant. 

Page B88, line 16 (10/22/20)

\hspace{1cm} The mode is temporarily set to zero while processing \texttt{\write} texts. 

Page B102, lines 3 and following of §241 (12/11/20)

\hspace{1cm} information, something special is needed. The program here simply assumes that suitable values appear in the global variables \texttt{sys_time}, \texttt{sys_day}, \texttt{sys_month}, and \texttt{sys_year} (which are initialized to noon on 4 July 1776, in case the implementor is careless). 

\hspace{1cm} \textbf{procedure fix_date_and_time;}
\hspace{1cm} \hspace{1cm} \textbf{begin sys_time} \leftarrow 12 \times 60; \textit{sys_day} \leftarrow 4; \textit{sys_month} \leftarrow 7; \textit{sys_year} \leftarrow 1776; \{ \textit{self-evident truths}\}
\hspace{1cm} \hspace{1cm} \hspace{1cm} \textit{time} \leftarrow \textit{sys_time}; \{ \textit{minutes since midnight}\}
\hspace{1cm} \hspace{1cm} \hspace{1cm} \textit{day} \leftarrow \textit{sys_day}; \{ \textit{day of the month}\}
\hspace{1cm} \hspace{1cm} \hspace{1cm} \textit{month} \leftarrow \textit{sys_month}; \{ \textit{month of the year}\}
\hspace{1cm} \hspace{1cm} \hspace{1cm} \textit{year} \leftarrow \textit{sys_year}; \{ \textit{Anno Domini}\}
\hspace{1cm} \hspace{1cm} \textbf{end;}

Page B103, replacement for §246 (12/11/20)

\hspace{1cm} \textbf{246.} Of course we had better declare a few more global variables, if the previous routines are going to work.

\hspace{1cm} \langle \text{Global variables } \texttt{13} \rangle + \equiv \langle \text{old setting: } 0 \ldots \texttt{max_selector}; \texttt{sys_time, sys_day, sys_month, sys_year: integer}; \{ \text{date and time supplied by external system}\} \rangle
The enclosing \{ and \} characters of a macro definition are omitted, but an output routine will be enclosed in braces.

routines that should be aborted, but we can sketch the ideas here: For a runaway definition or a runaway balanced text, we will insert a right brace; for a runaway preamble, we will insert a special \cr token and a right brace; and for a runaway argument, we will set long_state to outer_call and insert \par.

function str_toks(b : pool_pointer): pointer;  { converts str_pool[b .. pool_ptr - 1] to a token list }

begin continue: get_token;  { set cur_cmd, cur_chr, cur_tok }

if cur_tok < left_brace_limit then

help2("I'm going to ignore the \# sign you just used,")
("as well as the token that followed it."); error; goto continue;

help1("This \read has unbalanced braces."); align_state ← 1000000; limit ← 0; error;

494. Here is a procedure that ignores text until coming to an \or, \else, or \fi at the current level of \if \ldots \fi nesting. After it has acted, cur_chr will indicate the token that was found, but cur_tok will not be set (because this makes the procedure run faster).
command is being processed. Beware: For historic reasons, this code foolishly conserves a tiny bit of string pool space; but that can confuse the interactive `E` option.

if name = str_ptr − 1 then { conserve string pool space (but see note above) }

so-called boundary character of this font; the value of next_char need not lie between be and ec. If the very last instruction of the lig_kern array has skip_byte = 255, there is a special ligature/kerning program for a boundary character at the left, beginning at location 256 * op_byte +

Each portion of a formula is classified as Ord, Op, Bin, Rel, Open, Close, Punct, or Inner, for

begin char_warning(cur_f, qo(cur_c)); math_type(a) ← empty; cur_i ← null_character;

\[ s \leftarrow \text{fraction_noad} \]

\[ \text{cur_loop} \leftarrow \text{link(cur_loop)}; \text{link}(p) \leftarrow \text{new_glue(glue_ptr(cur_loop))}; \text{subtype(link}(p)) \leftarrow \text{tab_skip_code} + 1; \]

\[ \text{stat if tracing_paragraphs} > 0 \text{ then end_diagnostic(true); tats} \]

\[ \text{stat if tracing_paragraphs} > 0 \text{ then begin_diagnostic; tats} \]

hn: 0 .. 64; { the number of positions occupied in hc; not always a small_number }

The entry `height, §981.` here and on many later odd-numbered pages should be `height = macro, §135.`

to be in the range \( a \leq x \leq b \). System error messages should be suppressed when undumping.
loop. (Actually there’s one way to get error messages, via prepare_mag; but that can’t cause
infinite recursion.)
If final_cleanup is bypassed, this program doesn’t bother to close the input files that may still
be open.

begin (Finish the extensions 1378); new_line_char ← −1;

begin c ← cur_chr; if c ≠ 1 then new_line_char ← −1;

begin clear_terminal;
loop

begin goto breakpoint;
   { go to every declared label at least once }
breakpoint: m ← 0; @(‘BREAKPOINT’@)

they occupy in a typical production system (executable code size for dark blocks, global data
size for light blocks). In this way the chart indicates a total of about 12 × 22 = 264K bytes of
memory, plus 12 × 10 = 120K for the dynamic memory region not shown explicitly. The dynamic
memory is often considerably larger in practice, because it is desirable to accommodate large
macro packages and large pages.

20 More About Macros

that has already been designed. All you’ll see is ‘(io.mf The letter O [79])’ or possibly only ‘(io.mf [79])’, followed by ‘*’. Now the fun starts: You should type

uniformdeviate -100 -36.1628
z slanted 1/6 (0.16667y+x,y)
(a,b)zscaled(3,4) (-4b+3a,3b+4a)
(a,b)zscaled dir 30 (-0.5b+0.86603a,0.86603b+0.5a)
(a,b)dotprod(3,4) 4b+3a
Page C72, lines 4–18

⟨numeric atom⟩ → ⟨numeric variable⟩
| ⟨numeric token primary⟩
| ⟨(numeric expression)⟩
| normaldeviate
| length ⟨string primary⟩
| length ⟨path primary⟩
| length ⟨pair primary⟩
| angle ⟨pair primary⟩
| xpart ⟨pair primary⟩
| ypart ⟨pair primary⟩
| ⟨numeric operator⟩⟨numeric primary⟩

⟨numeric token primary⟩ → ⟨numeric token⟩ / ⟨numeric token⟩
| ⟨numeric token not followed by '⟩ ⟨numeric token)'⟩

⟨numeric primary⟩ → ⟨numeric atom not followed by ⟕ ⟨expression⟩ , ⟕⟩
| ⟨numeric atom⟩ ⟕ ⟨numeric expression⟩ , ⟨numeric expression⟩ ⟕

Page C76, lines 8–16 from the bottom

tom edge of the type. (With plain METAFONT’s beginchar each character has a “bounding box” that runs from (0, h) at the upper left and (w, h) at the upper right to (0, −d) and (w, −d) at the lower left and lower right; variable d represents the depth of the type. The values of w, h, and d might change from character to character, since the individual pieces of type need not have the same size in a computer-produced font.)

Page C80, line 14

penpos ⟨suffix⟩ ⟨⟨unknown⟩, ⟨known⟩⟩.

Page C83, line 16

### 0.5a=-c-0.5b+1.5

Page C83, line 19

the only dependent variable is now d, which equals 0.5c + 0.75b + 0.75. (This is

Page C96, line 13 from the bottom

illustrates the use of u#, s#, hi#, logo_pen, leftstemloc, o, xgap, and barheight:
Page C106, lines 19–21 (07/03/20)

pixels. (Some typesetting systems use both of these device-dependent amounts to alter
their current position on a page, just after typesetting each character. Other systems,
like typical \texttt{dvi} software associated with \TeX{}, assume that \texttt{chardy} = 0 but use \texttt{chardx}

Page C113, lines 5–11 from the bottom (07/20/20)

\begin{verbatim}
s# := 5pt; define_pixels(s); % side of the square
z_1 = (0, 0); z_2 = (s, 0); z_3 = (0, s); z_4 = (s, s);
for k = 1 upto 4: z[k + 4] = z[k] + (\frac{s}{2}, \frac{s}{2}); \textbf{endfor}
pickup pencircle scaled 4pt; \textbf{draw} z_5 \ldots z_8 \ldots cycle;
pickup pencircle scaled 1.6pt; \textbf{erase draw} z_2 \ldots z_4 \ldots z_3;
pickup pencircle scaled 4pt; \textbf{draw} z_1 \ldots z_2 \ldots z_4 \ldots z_3 \ldots cycle;
for k = 1 upto 4: \textbf{draw} z[k] -- z[k + 4]; \textbf{endfor}.
\end{verbatim}

Page C114, line 7 (07/20/20)

\begin{verbatim}
for k = 0 upto 4: z[k] = center + (radius, 0) rotated(90 + \frac{360}{5} k); \textbf{endfor}
\end{verbatim}

Page C128, lines 13 and 14 (06/13/20)

changed. Plain \textsc{Metafont} has a \texttt{tensepath} operation that does this. For example,
\texttt{tensepath} \texttt{unitsquare} = (0, 0) \ldots (1, 0) \ldots (1, 1) \ldots (0, 1) \ldots cycle.

Page C136, lines 18 and 19 (07/17/20)

only about 0.28 with respect to the initial and final directions; since \textsc{Metafont} insists
that tensions be at least 0.75, this anomalous path could never have arisen if the control

Page C155, line 7 (10/07/20)

\langle \texttt{program} \rangle \rightarrow \langle \texttt{statement list} \rangle \langle \texttt{statement} \rangle \texttt{end}

Page C160, lines 7–9 (06/25/20)

might produce a transcript that includes the following diagnostic information:

\begin{verbatim}
rotatedaround(EXPR0)(EXPR1)--> shifted(EXPR0)rotated(EXPR1)shifted(EXPR0)
\end{verbatim}

Page C165, lines 5–7 from the bottom (11/11/17)

(i.e., parameters in parentheses), then we name zero or one or two undelimited param-
eters. Then comes an `=\texttt{=}` sign, followed by the replacement text, and \texttt{enddef}. The
`=\texttt{=}` sign might also be `:\texttt{=}': both mean the same thing.

Page C171, lines 18–20 (08/16/20)

Chapter 14’s syntax rules for \langle \texttt{path primary} \rangle, via \langle \texttt{pair primary} \rangle. A pair ex-
pression is not considered to be of type \texttt{path} unless the path interpretation is the
only possibility.
Page C176, line 7 from the bottom (07/09/20)

if @#(x_): tx := f_x_ fi := x_; endfor

Page C180, line 3 from the bottom (06/24/20)

'=' or ':=' following let.

Page C187, line 11 from the bottom (07/12/20)

| substring (pair expression) of (string primary)

Page C189, line 14 (06/13/20)

'! ' and followed by ',', followed by lines of context as in METAFONT's normal error

Page C200, line 12 from the bottom (08/27/20)

\[ y_1 = y_2 = \text{good.y.(5[-d,h] + 1.1pt)}; \]

Page C202, line 17 from the bottom (06/13/20)

command, and it works only when the penpos angle is 0. If the penpos command is

Page C210, bottom eight lines, and top ten lines of page C211 (07/16/20)

\[(\text{numeric atom}) \rightarrow (\text{numeric variable}) \mid (\text{numeric argument}) \mid (\text{numeric token primary}) \mid (\text{internal quantity}) \mid \text{normaldeviate} \mid (\text{numeric expression}) \mid \text{begingroup} \langle \text{statement list} \rangle \langle \text{numeric expression} \rangle \text{endgroup} \mid \text{length} \langle \text{numeric primary} \rangle \mid \text{length} \langle \text{pair primary} \rangle \mid \text{length} \langle \text{path primary} \rangle \mid \text{length} \langle \text{string primary} \rangle \mid \text{ASCII} \langle \text{string primary} \rangle \mid \text{oct} \langle \text{string primary} \rangle \mid \text{hex} \langle \text{string primary} \rangle \mid \langle \text{pair part} \rangle \langle \text{pair primary} \rangle \mid \langle \text{transform part} \rangle \langle \text{transform primary} \rangle \mid \text{angle} \langle \text{pair primary} \rangle \mid \text{turningnumber} \langle \text{path primary} \rangle \mid \text{totalweight} \langle \text{picture primary} \rangle \mid \langle \text{numeric operator} \rangle \langle \text{numeric primary} \rangle \mid \text{directiontime} \langle \text{pair expression} \rangle \mid \text{of} \langle \text{path primary} \rangle \mid \langle \text{numeric token primary} \rangle \rightarrow \langle \text{numeric token} \rangle \langle \text{numeric token} \rangle \mid \langle \text{numeric token not followed by '}' \rangle \langle \text{numeric token} \rangle \mid \langle \text{numeric primary} \rangle \rightarrow \langle \text{numeric token not followed by [''] \rangle \langle \text{expression} \rangle \mid \langle \text{numeric atom} \rangle \langle \text{numeric expression} \rangle \langle \text{numeric expression} \rangle \]

Page C214, line 6 becomes two lines (07/17/20)

\[(\text{future pen primary}) \rightarrow (\text{future pen argument}) \mid \text{pencircle} \]
Page C214, line 6 from the bottom (07/12/20)

| substring (pair expression) of (string primary)

Page C217, lines 20–25 (10/07/20)

⟨program⟩ → ⟨statement list⟩⟨non-title statement⟩ end
| ⟨statement list⟩⟨non-title statement⟩ dump
⟨statement list⟩ → ⟨empty⟩ | ⟨statement⟩ ; ⟨statement list⟩
⟨statement⟩ → ⟨empty⟩ | ⟨title⟩
| ⟨equation⟩ | ⟨assignment⟩ | ⟨declaration⟩
| ⟨definition⟩ | ⟨compound⟩ | ⟨command⟩

Page C219, line 25 (05/25/20)

to see which of its subscripts and suffixes have occurred. For example, if you're

Page C224, lines 7–9 from the bottom (12/21/18)

y₄r=-0.9848thinn+259.00049
x₄r=-0.08682thinn+144
y₄=-0.4924thinn+259.00049

Page C226, lines 9 and 10 (11/01/20)

This means that the preloaded base you have specified cannot be used, because it is corrupted or was prepared for a different version of METAFONT.

Page C228, line 27 (06/19/20)

1.94 endfor

Page C228, line 4 from the bottom (07/12/20)

might want to review now.) You probably also have a proof mode diagram:

Page C234, line 4 of answer 4.6 (07/20/20)

for k = 1 upto 6: \[ z[k]' = 2[z[k], z₀]; \] endfor

Page C241, line 2 (11/11/17)

\mode=cheapo; input cheaplogo10

Page C242, line 11 of answer 13.7 (07/20/20)

for k = 1 upto 4: \[ z[k + 4] = z[k] + (\frac{2}{3}s, \frac{1}{3}s); \] endfor
**Page C243, lines 7 and 8**

(11/08/15)

```
draw subpath(\(k, k + 1\)) of \textit{star}; \texttt{cullit};
undraw subpath(\(k + 2, k + 3\)) of \textit{star withpen eraser}; \texttt{cullit};
```

**Page C243, line 3 of answer 13.11**

(06/17/20)

```
def overdraw expr \(c = \texttt{begingroup save region;}\)
```

**Page C243, lines 12–16 of answer 13.11**

(05/24/20)

```
beginchar(\texttt{("M",1.25in\#,,.5in\#,0)}); \texttt{pickup pencircle scaled .4pt};\(z_1 = (20,-13); z_2 = (30,-6); z_3 = (20,1); z_4 = (4,-7);\)
\(z_5 = (-12,-13); z_6 = (-24,-4); z_7 = (-15,6);\)
\texttt{path M; M = (\textit{origin} \ldots z_1 \ldots z_2 \ldots z_3 \ldots z_4 \ldots z_5 \ldots z_6 \ldots z_7 \ldots \textit{origin} \ldots -z_7 \ldots -z_6 \ldots -z_5 \ldots -z_4 \ldots -z_3 \ldots -z_2 \ldots -z_1 \ldots \textit{cycle});}
```

**Page C246, line 2 of answer 14.13**

(08/16/20)

path \(z_0 \ldots z_1\) is equivalent to \('z_0 \ldots \texttt{controls 1/3[z_0,z_1]} \texttt{and 2/3[z_0,z_1]} \ldots z_1',\) and the

**Page C247, line 1 of answer 15.5**

(06/13/20)

```15.5. \texttt{beginchar(126,25u\#,h\texttt{height\#} + \texttt{border\#},0); "Dangerous left bend";}```

**Page C247, replacement for answer 15.7**

(07/21/20)

```
15.7. Replace lines 10 and 11 by
    \texttt{pickup pencircle scaled 3/4pt yscaled 1/3 rotated \texttt{-60};}
    \texttt{draw (z_1 \ldots p) transformed t;}
    \texttt{addtto currentpicture also currentpicture}
    \texttt{rotatedaround(\((.5w,.5h)\) yscaled \texttt{aspect\_ratio}, \texttt{-180});}
```

**Page C249, line 1 of answer 18.9**

(08/02/20)

```
18.9. \texttt{beginchar ("H",13u\#,"ht"\#,0); \texttt{pickup broad\_pen;}}
```

**Page C249, line 11 of answer 18.9**

(08/02/20)

```
\texttt{filldraw bot\_serif\_edge_4}
```

**Page C250, line 4 of answer 19.1**

(04/19/20)

because it saves a wee bit of time and because \texttt{;} \texttt{often belongs before endfor.}

**Page C250, replacement for answer 19.3**

(07/12/20)

```
19.3. Yes, if and only if \(n - \frac{1}{2}\) is an even integer. (Because ambiguous values are rounded upwards.)
```
22.1 (a) If and only if $n$ is an integer between 0 and 255. (b) If and only if $s$ is a string of length 1.

Page C254, lines 10–13 from the bottom become five lines (06/26/20)

? H I found no right delimiter to match a left one. So I've put one in, behind the scenes; this may fix the problem.

? 

Page C260, the “line” after line 3 (06/14/20)

\[
\begin{align*}
\text{font size} & \\
\text{font slant} & \\
\text{font normal space} & \\
\text{font normal stretch} & \\
\text{font shrink} & \\
\text{font x height} & \\
\text{font quad} & \\
\text{font extra space} & \\
\end{align*}
\]

\[
\begin{align*}
\{ & \\
\{ & \\
\{ & \\
\{ & \\
\{ & \\
\end{align*}
\]

Page C261, lines 16 and 17 from the bottom (06/14/20)

\[
\begin{align*}
\text{proofrule} & \\
\text{screenrule} & \\
\text{makegrid} & \\
\text{proofrulethickness} & \\
\text{prooffoffset} & \\
\end{align*}
\]

Page C266, lines 19 and 20 (07/04/20)

You can say either ‘incr x’ or ‘incr (x)’, within an expression; but neither of them are valid statements by themselves.

Page C269, line 11 (01/10/21)

\[
\text{\textbackslash mmode=} \text{specmode}; \text{mag=} \text{magnification}; \text{input} \{\text{font file name}\}
\]

Page C277, lines 15–19 (03/06/17)

def openit = openwindow currentwindow from origen % and please correct to \(\text{screen_rows,screen_cols}\) at (-50,300) enddef; % "(-50,300)" too
def showit_ = display currentpicture inwindow currentwindow enddef;
def showit = openit; let showit=showit_; showit enddef; % first time only

Plain \texttt{METAFONT} has several other terse commands similar to ‘openit’ and ‘showit’:

Page C279, line 1 (11/11/17)

\[
\text{blacker=} .1; \quad \% \text{ make pens a teeny bit blacker}
\]
Page C289, line 20 (10/07/20)

    if {{(pair x) cand x>(0,0)}}: A else: B fi.

Page C291, line 18 (07/24/20)

    save u_; setu_ u; let switch_ = if; if false: endif.

Page C292, line 10 from the bottom (10/23/20)

be known by saying `if known (p−q): p = q else: false fi'; transforms could be handled

Page C293, lines 13 and 14 from the bottom (10/27/20)

>f(−1) is false! When c → 0, the quantity a³ + b³ approaches −∞ when c is positive,
+∞ when c is negative. An attempt to `solve f(1,−1)' will divide by zero and come

Page C295, line 2 (07/04/20)

>‘interpolate (1,1) .. (3,2) .. (15,4) of 7’ the approximate value 3.37.

Page C299, bottom four lines of code become five (08/06/20)

    primarydef t Bernstein nn = begingroup save r; r =
    begingroup for n=nn downto 2:
    for k=1 upto n-1: u_[[k]]:=t([[u_[[k]]],u_[[k+1]]]] ]]];
    endfor endfor u_[[1]]] endgroup; numeric u_[[1]]];
    r endgroup enddef;

Page C299, line 5 after the code becomes two lines (08/06/20)

brackets are nested inside of brackets. However, the auxiliary variables 'u_[[[k]]]' must not remain independent at the end.

Page C305, lines 14–18 (07/08/20)

width_adj#:=#0pt#; % width adjustment for certain characters
serif_fit#:=#0pt#; % extra sidebar near lowercase serifs

; 
low_asterisk:=false; % should the asterisk be centered at the axis?
math_fitting:=false; % should math-mode spacing be used?

Page C317, line 21 becomes two lines (11/11/17)

⟨label⟩ −→ ⟨code label⟩ | ⟨code⟩ :: | :
⟨code label⟩ −→ ⟨code⟩ ;
Page C318, lines 10–16 from the bottom (11/11/17)

| ⟨code label⟩⟨labeled code⟩
{extensible command} → extensible ⟨code label⟩{four codes}
{four codes} → ⟨code⟩, ⟨code⟩, ⟨code⟩, ⟨code⟩

Notice that a ⟨code label⟩ can appear in a ligtable, charlist, or extensible command. These appearances are mutually exclusive: No code may be used more than once as a label. Thus, for example, a character with a ligature/kerning program cannot also be extensible, nor can it be in a charlist (except as the final item).

Page C333, line 29 (10/25/19)

"if charcode>0:currentpicture:=currentpicture scaled mg;fi;"

Page C333, bottom two lines become one (11/11/17)

if unknown scale: scale := max(1,round(pixels_per_inch/300)); fi

Page C339, line 3 (05/21/20)

ing ‘ß’, ‘æ’, ‘œ’, and ‘ø’) and the uppercase letters (including ‘Æ’, ‘Œ’, and ‘Ø’) are

Page C341, line 14 from the bottom (11/11/17)

prints the \table and the \text; \bigtest gives you the works, plus a mysterious word

Page C345 and following, selected amendments to the index (01/20/21)

(addto command), 118, 220.
bell-shaped distribution, 253, 251.
black, 270, 332–333.
⟨code⟩ and ⟨code label⟩, 317.
concatenation, of paths, 70–71, 123, 127–129, 130, 137, 245, 266.
of strings, 69, 73, 84–85, 187, 278, 286, 312.
*directiontime, 135, 136, 211, 245, 265, 298.
distance, 76, 84, see also length.
dotprod, 68–69, 178, 238, 260.
efficiency, 39, 99, 116, 141, 144, 147, 228, 230, 234, 244, 264, 265, 277, 291, 297, 298.
empty option in for list, 171, 172, 299.
forbidden tokens, 173, 218–219, 286.
*fread, 191, 220, 252, 277, 312.
Giotto di Bondone, 139.
independent variables, 81–83, 88, 224, 226, 299.
\init, 337, 342.
internal quantities, 54–55, 88, 218, 262, 265–266.
*invwindow, 191, 220, 277.
(keep or drop), 118, 220.
labels, 107, 274, 327–328.
*length, 66, 69, 72, 210, 238.


N, 184–185, 302–303.

⟨numeric token primary⟩, 72, 211.

o, 23, 34, 93, 197, 200, 204, 240, 302.


overshoot, 23, 34, 93, 197, 200, 204, 302.

pens, 26–29, 37, 80, 103, 162, 273, 310.

r, 21–29, 147–152, 297–298.

*rotated, 21–22, 25, 27, 44, 68, 73, 107, 114, 117, 141, 213, 238.

rule, 274, 326.

*scaled, 21–23, 68, 73, 141, 213, 244, 291.

*showstopping, 211, 219, 227, 230, 262.

string expressions, 69, 187–189, 258, 286.

(suffix list), 171, 236.

sum, of vectors, 9, 68.

test.mf, 311–313.


text arguments, 219, 288–291, 299.

.TFM, 39, 315–321, 333, 335.

*to, 191, 220, 252, 277, 312.

undelimited suffix parameters, 167, 176, 266, 270.

undraw, 113, 118, 120, 242, 271.

unitsquare, 116, 123–124, 125, 132, 136, 263.

*unknown, 170, 210.

unknown quantities, nonnumeric, 84–85, 143.


(vardef heading), 165, 178.

*xscaled, 21–22, 68, 73, 141, 213, 244, 291.

Page Dv, line 16 (01/16/21)

I believe that the final bug in METAFONT was discovered on January

Page Dv, bottom two lines (01/16/21)

corporates all of those changes. I now believe that the final bug was discovered on 03 July 2020 and removed in version 2.71828182. The finder’s fee has converged to $327.68.

Page D2, last line of §2 (01/15/21)

define banner ≡ 'This is METAFONT, v.Version_2.71828182' { printed when METAFONT starts }

Page D14, line 1 of §30 (05/05/14)

30. The input/in function brings the next line of input from the specified file into available

Page D21, line 8 of §47 (10/11/20)

g: str_number; { the string just created }

Page D27, lines 3 and 4 of §61 (04/02/17)

is not serious since we assume that this part of the program is system dependent.
Page D28, line 7 (04/02/17)

\[ \text{var } k: 0 \ldots 23; \{ \text{index to current digit; we assume that } |n| < 10^{23} \} \]

Page D32, line 2 of §78 becomes two lines (06/27/20)

\[
\text{loop begin continue: if interaction } \neq \text{ error_stop_mode then return;}
\]
\[
\text{clear_for_error_prompt: prompt_input("?"};
\]

Page D32, line 11 of §79 (07/03/20)

\[ "E": \text{if file_ptr > 0 then if input_stack[file_ptr].name_field } \geq 256 \text{ then} \]

Page D33, line 5 of §80 (07/03/20)

\[ \text{if file_ptr } > 0 \text{ then if input_stack[file_ptr].name_field } \geq 256 \text{ then print("E_to_edit_your_file."} \]

Page D37, line 9 of §93 (08/07/20)

\[ (\text{"Try_to_insert_an_instruction_for_me_e.g.,_’I_show_a;’},\text{)} \]

Page D82, line 2 from the bottom (09/19/19)

\[ \text{define boundary_char } = 41 \{ \text{the boundary character for ligatures} \} \]

Page D85, lines 3 and 4 of §194 (and §194 actually moves to page D86) (12/11/20)

information, something special is needed. The program here simply assumes that suitable values appear in the global variables \text{sys_time}, \text{sys_day}, \text{sys_month}, and \text{sys_year} (which are initialized to noon on 4 July 1776, in case the implementor is careless).

Page D85, the final six lines of §194 (and §194 actually moves to page D86) (12/11/20)

\[ \text{procedure fix_date_and_time; begin sys_time } \leftarrow 12 \times 60; \text{ sys_day } \leftarrow 4; \text{ sys_month } \leftarrow 7; \text{ sys_year } \leftarrow 1776; \{ \text{self-evident truths} \}
\]
\[ \text{internal}[\text{time}] \leftarrow \text{sys_time } \times \text{unity}; \{ \text{minutes since midnight} \}
\]
\[ \text{internal}[\text{day}] \leftarrow \text{sys_day } \times \text{unity}; \{ \text{day of the month} \}
\]
\[ \text{internal}[\text{month}] \leftarrow \text{sys_month } \times \text{unity}; \{ \text{month of the year} \}
\]
\[ \text{internal}[\text{year}] \leftarrow \text{sys_year } \times \text{unity}; \{ \text{Anno Domini} \}
\]
\[ \text{end;} \]

Page D86, replacement for §196 (12/11/20)

\[ 196. \text{ Of course we had better declare a few more global variables, if the previous routines are going to work.} \]
\[ \{ \text{Global variables } 13 \} + \equiv \]
\[ \text{old_setting}: 0 \ldots \text{max_selector}; \text{sys_time, sys_day, sys_month, sys_year: integer}; \{ \text{date and time supplied by external system} \} \]

Page D97, line 2 of §221 (05/26/17)

the definition of attribute nodes) that it is convenient to let \text{info}(p) = 0 stand for ‘\text{[]}’.
but the log \( n \) factor is buried in our implicit restriction on the maximum raster size.)

\[
\begin{align*}
\text{Page D237, line 5 of §513} & \quad (05/26/17) \\
\text{for } n \leftarrow 0 \text{ to } n1 - n0 - 1 \text{ do } env_{move}[n] \leftarrow mm0;
\end{align*}
\]

direction \((right_u(p), left_v(q))\); and there's a line of length \( \geq \delta \) from vertex \( q \) to vertex \( r \),

\[
\begin{align*}
\text{Page D296, line 11} & \quad (06/23/20) \\
\text{name} \text{ points to the } eqtb \text{ address of the macro being expanded, if the current token list}
\end{align*}
\]

\[
\begin{align*}
\text{Page D324, line 13 of §713} & \quad (12/20/20) \\
\text{help2("After \text{'exitif',<boolean_expr>' I expect to see a semicolon."\)}}
\end{align*}
\]

\[
\begin{align*}
\text{Page D326, line 5 from the bottom} & \quad (06/23/20) \\
\{ \text{invokes a user-defined sequence of commands} \}
\end{align*}
\]

\[
\begin{align*}
\text{Page D334, lines 1 and 2 of §742} & \quad (10/25/20) \\
\text{742. Here is a procedure that ignores text until coming to an elseif, else, or fi at the current level of if...fi nesting. After it has acted, } cur_{mod} \text{ will indicate the token that was found.}
\end{align*}
\]

\[
\begin{align*}
\text{Page D339, line 4 of §757} & \quad (06/16/20) \\
\text{(A user who tries some shenanigan like } \text{'for ... let endfor'} \text{ will be foiled by the get_symbol}
\end{align*}
\]

\[
\begin{align*}
\text{Page D351, lines 2-7 of §790 become five lines} & \quad (12/11/20) \\
\text{begin } wlog(banner); \text{ slow_print(format_ident); print("\_\_\_"); print_int(sys\_day); print_char("\_\_\_");} \\
\text{months } \leftarrow \text{'JANFEBMARAPRILMAYJUNJULAUGSEPTEMBNOVDEC';} \\
\text{for } k \leftarrow 3 \ast \text{sys\_month} - 2 \text{ to } 3 \ast \text{sys\_month} \text{ do } wlog(months[k]);} \\
\text{print_char("\_\_\_"); print_int(sys\_year); print_char("\_\_\_"); print_two(sys\_time div 60); print_char("\_\_\_");} \\
\text{print_two(sys\_time mod 60);}
\end{align*}
\]

\[
\begin{align*}
\text{Page D352, line 2 of §793 becomes two lines} & \quad (10/29/20) \\
\text{command is being processed. Beware: For historic reasons, this code foolishly conserves a tiny bit of string pool space; but that can confuse the interactive 'E' option.}
\end{align*}
\]

\[
\begin{align*}
\text{Page D352, line 5 from the bottom} & \quad (10/29/20) \\
\text{if } name = str\_ptr - 1 \text{ then } \{ \text{conserve string pool space (but see note above)} \}
\end{align*}
\]
cur_type = path_type means that cur_exp points to the first node of a path; nobody else points

doing a similar thing in the rare case that a boundary character is the last character of a

so-called boundary character of this font; the value of next_char need not lie between bc and ec.
If the very last instruction of the lig_kern array has skip_byte = 255, there is a special
ligature/kerning program for a boundary character at the left, beginning at location 256 * op_byte +

tional halt; no ligature or kerning command is performed.

Page D471, lines 20 and 21

param: array [1 .. max_font_dimen] of scaled; { fontdimen parameters}
np: 0 .. max_font_dimen; { the largest fontdimen parameter specified so far}

Page D474, line 2 from the bottom

help1("A colon should follow a headerbyte or fontdimen location."); back_error;

Page D508, line 3 of §1189.

to be in the range $a \leq x \leq b$. System error messages should be suppressed when undumping.

Page D516, line 16

If final_cleanup is bypassed, this program doesn’t bother to close the input files that may still
be open.

Page D519, line 17

fix_date_and_time; init_randoms(sys_time + sys_day * unity);

Page D520, line 18 of §1212 becomes two lines

begin clear_terminal;
loop

Page D520, lines 11 and 12 from the bottom of §1212 become three lines

begin goto breakpoint;
{ go to every declared label at least once }
breakpoint: m ← 0; @{‘BREAKPOINT’@}

Page D566, the bottom five lines

they occupy in a typical production system (executable code size for dark blocks, global data
size for light blocks). In this way the chart indicates a total of about $8 \times 22 = 176K$ bytes of
memory, plus $8 \times 15 = 120K$ for the dynamic memory region not shown explicitly. The dynamic
memory is often considerably larger in practice, because it is desirable to accommodate large
macro packages and large pictures.