one 1.2 MB diskette or three 360KB diskettes. No subsetting, please. Send David

1. Formatted diskettes,
2. Indication of the format required,
3. A self-addressed mailer, and
4. A $5.00 donation per set of files, to cover postage and equipment wear & tear. (If you live outside North America, airmail delivery will probably require more postage. You should probably contact David for details.)

NB: 1. David does not have the CTEX files. 2. No phone calls or personal visits.

David's address:

David W. Hopper
446 Main Street
Toronto, Ontario
Canada M4C 4Y2
Thanks, David.

Editor's note: Traffic on the network servers and gateways has been very high recently, and in order to provide improved service, there have been some volunteers to maintain local "slave" repositories of the LATEX style collection. There is usually a geographic or network restriction requested, since the idea is to cut down traffic, not add to it. The following areas will be covered by the volunteers listed.

- Bitnet users: Texas A&M maintains a list- and file-server which is already handling (with TEX-L) much of the Bitnet distribution of TEXhax. An inquiry via listserv will retrieve a list of all TEX-related files:
  `tell listserv at tamvm1 get tex filelist`
- United Kingdom, for users of JANET or uucp: Stephen Page, `sdpage@uk.ac.ox.prg` or `...!ukc!ox-prg!sdpage`
- European users of BITnet: Christoph Gatzka, `zrgc002@dtuzdv5a.Bitnet`

Additional volunteers should contact Ken.

The LATEX User's Column

Jackie Damrau
University of New Mexico

Since there has not been much activity in this column, we have decided to try to take problems and answers from TEXhax. If anyone who submits material to TEXhax does not want an item republished, please let me know. The questions section below starts off with a few items from other mail networks.

At the TEX conference in August, there was some discussion that this column should also be a question and answer column for beginners. So **BEGINNERS**, please submit your questions to

Jackie Damrau
Dept. of Mathematics and Statistics
University of New Mexico
Albuquerque, New Mexico 87131
    (505) 277-4623

or

UUCP: `damrau@ariel.unm.edu`
Bitnet: `damrau@unmb.Bitnet`

A LATEX Bug

We did receive a copy of a letter to Leslie Lamport about a possible LATEX bug:

I recently uncovered a bug in LATEX Version 2.09 (as of April 19, 1986). While inside the `tabbing` environment, every time a `\>` command appears two words on TEX's save stack get used. Thus any large `tabbing` environment can cause a save stack overflow, resulting in this error message

`TeX capacity exceeded, sorry.`

This is especially true if your system is configured with a small save stack size. The problem stems from your doing both global and local assignments to box register `\@curline` and is explained in *The TEXbook* on page 301.

I fixed the problem by changing this declaration

```
\def\@addfield{\global\setbox\@curline
        \hbox{\unhbox\@curline
                \unhbox\@curfield}}
```

to

```
\def\@addfield{\setbox\@curline
        \hbox{\unhbox\@curline
                \unhbox\@curfield}}
```

Note the absence of the `\global` in the redefinition. The fix seems to work fine, but I don't know if it will cause any bad repercussions somewhere else.

I also caused LaTeX's output routine to go into an infinite loop, but I hesitate to call it a bug because I have only known it to occur with one of my document styles. The problem occurred when \output was being called just after a \begin{enumerate}. Inside the output routine, TeX was executing \par over and over again and not making any progress. With a little hunting, I found that LaTeX disables \par just after the beginning of list constructions. I also narrowed down the point where the infinite loop was occurring to this construction within the header text

```
\vbox to.4in{\vfil\hsize 4in ...

                some text ... \vfil}
```

When the second \vfil command got executed, TeX, for some reason, tried to execute \par. Since \par at this point was disabled, TeX kept trying.

My solution was to add this

```
\let\par=\@@par
```

to the beginning of the \output definition. Since, I believe, your document styles won't cause a similar construction to occur in the output routine, this change is not needed for standard LaTeX, but I believe it can help to make LaTeX more bullet-proof for users who can't leave well enough alone.

Many thanks for all your work in developing LaTeX. You have made my job of helping our staff produce TeX-quality documents immeasurably easier.

Benjamin J. Kennedy
Xybion Corporation

Questions from the mail networks

Here are some questions, answers and macros that appeared on the mail network. If anyone has answers to these questions, please send the answers to both the originator and myself.

Question 1. A problem I had with verbatim was to include a file inside a verbatim environment. I wanted to have a file of sample LaTeX code that could be LaTeXed alone correctly and then included verbatim as an appendix to another document. Of course, I could have copied the file in with my editor, but I wanted changes in the sample file to be automatically changed in the version in the appendix.

The problem, of course, is that

```
\begin{verbatim}
\input examplefile
\end{verbatim}
```

only prints \input examplefile. One can place the verbatim commands in the example file, but

then one can not use the example file directly to generate the example. A solution:

```
% necessary to use expandafter
\def\bv{\begin{verbatim}}
\def\infil{\input examplefile}
\expandafter\bv\infil
\end{verbatim}
```

There may be a similar solution with more \expandafters that avoids the \defs, but at least this worked.

Terry L. Anderson
AT&T Bell Laboratories – Liberty Corners
UUCP: ...!ihnp4!kaiser!tla

Editor's note: This solution can be extended to process multiple examples in the same document in the following manner.

```
\def\infil{\input\examplefile}
\def\bv{\begin{verbatim}}
\def\doexample#1{\def\examplefile{#1}%
        \expandafter\bv\infil}
\doexample{file1}
\end{verbatim}
\doexample{file2}
\end{verbatim}
```

As Terry Anderson said above, there may be a solution with more \expandafters that would avoid having to say \end{verbatim} explicitly for each file, but, ...

Question 2. I'm just a beginner with LaTeX, and I would like to define an environment of my own, including some verbatim. Something like this:

```
\newenvironment{myenv}
    {\begin{verbatim}}{\end{verbatim}}
\documentstyle{book}
\pagestyle{headings}
\begin{document}

\chapter{Hi}

Here it comes:
\begin{myenv}
Try and try
            ... again!
\end{myenv}

This is the end.
\end{document}
```

But it doesn't work. This is the most important part of the log file:

```
This is TeX, Version 2.0 for Berkeley UNIX
   (preloaded format=lplain 86.12.16)   23 Aug 87
(es.tex
LaTeX version 2.09 - Released 27 October 1986
(/usr/lib/tex/macros/book.sty
Document style 'book'. Released 4 September 1986
(/usr/lib/tex/macros/bk10.sty)
) (es.aux)
Chapter 1.
)
Runaway argument?

^^MTry and try^^M   ... again!^^M\end{myenv}^^MT\ET
! File ended while scanning use of \@xverbatim.
<inserted text>
                 \par
<*> es.tex

?
```

What happened?

Gérald Masini
Centre de Recherche en Informatique
de Nancy
uucp: masini@crin.crin.fr
CRIN, B.P. 239
54506 Vandoeuvre-les-Nancy Cedex
France

Editor's note: You get the same complaint about \@xverbatim when you include \end{verbatim} in the \doexample expansion shown in the note following Question 1. These problems are related. This doesn't explain why, though, and LaTeXperts are invited to provide an explanation scrutable to beginners.

Question 3. Tom Hoffman wrote a LaTeX macro that almost solves the .ip paragraph problem. [Editor's note: .ip is a troff macro indicating a labeled paragraph.] I found his mixture of TeX and LaTeX confusing, so I rewrote his solution in LaTeX:

```
\newenvironment{itemlist}[1]
  {\begin{list}{}
   {\settowidth{\labelwidth}{#1}
    \setlength{\leftmargin}{\labelwidth}
    \addtolength{\leftmargin}{\itemindent}
    \addtolength{\leftmargin}{\labelsep}
    \renewcommand{\makelabel}[1]
         {##1\hfill}}}%
  {\end{list}}
```

I also took a whack at a complete solution.

```
% Back to TeX we go.
\def\ml#1{%
        \newbox \newlabel
  \setbox \newlabel =\hbox{#1}%
     \ifdim \wd\newlabel >\labelwidth
        \newdimen \pad \pad=\textwidth
        \advance \pad by-\rightmargin
        \advance \pad by-\itemindent
        \advance \pad by-\labelsep
        \hbox to\pad {#1\hfill}%
     \else #1\hfill \fi}
\newenvironment{itemlist}[1]
  {\begin{list}{}
   {\settowidth{\labelwidth}{#1}
    \setlength{\leftmargin}{\labelwidth}
    \addtolength{\leftmargin}{\itemindent}
    \addtolength{\leftmargin}{\labelsep}
    \let\makelabel=\ml}}%
  {\end{list}}
```

This seems to work; it even nests, but the inner itemlists cause overfull \hbox errors. When I find I'm writing macros like \ml I usually stop, calm down, and look for another solution. However, I'm going to cut my losses and let this one go. Here's a hint for anybody who picks it up: white-space is very important (which is partly why I'm letting it go).

clayton@thumper.UUCP

Question 4. I am trying to use \vfill within the LaTeX table environment without much success. What I want to be able to do is to move the caption down to the bottom of the page using the rubber length \vfill. LaTeX seems to ignore the \vfill command. Am I missing something? Is there another way of doing the same thing?

My LaTeX file looks something like this:

```
\documentstyle[12pt]{report}
\begin{document}
\begin{table}[p]
\centerline{The actual table will go here}
\vfill
\caption[Test table]{This is the caption
 for the test table. It will contain
 a description of the salient features
 of the table.}
\end{table}
\end{document}
```

The \vfill command doesn't do anything. The caption stays 10pt below the table (or in this case the \centerline stuff). REP12.STY defines a vskip

of 10pt in its definition of @makecaption and I couldn't figure out how to change it to a rubber length. I also tried putting \caption in a parbox but that did the same thing.

Atul Kacker
...seismo!rochester!ur-tut!akk2
or akk2@tut.cc.rochester.edu

Become a contributor!

Send in your answers to these questions. And if you create anything else you find useful, please let others share in your new found knowledge.

Queries

Title formatting macro wanted

I have a peculiar problenm that seems made for TEX's typesetting capabilities. The problem is the following:

Given an input list consisting of words separated by spaces (not control sequences), produce a title block of entirely capitalized words, double-spaced, centered, and (gasp!) in inverted pyramid form. I would like the lines produced in general not to be of vastly differing lengths (no long lines followed by singletons, please), and I would prefer that line breaks in titles, if the rest of the criteria are met, occur at commas in preference to just being between words.

I admit that this may be a very tall order, but as a newcomer to TEX I have quickly come to realize the vast power of this typesetting system and can only hope that such a macro already exists. If I have placed too many limitations on the macro as it stands, please inform me of any solutions to the "inverted pyramid" problem and I will fiddle with them as best as possible to produce what I need.

I am using LATEX, if that is of any help in finding an "inverted pyramid" macro.

Stephen C. Lipp
University of Texas at Austin

Reply: Printing Out Selected Pages

Donald Knuth's reply to Helen Horstman in TUGboat Vol. 8, No. 2 (p. 217, "Print Out Selected Pages") contains a bug which can cause the omission of pages from the output, even if they were specified in the pages.tex file. Specifically, if the page number of the first page processed is greater than the first requested page, then no pages will be printed.

Here is an example:

```
\count0=2
This is the text for page 2.
\vfil\eject
This is the text for page 3.
\bye
```

If the pages.tex file contains the lines

```
1
3
```

the first page number will never be matched, and no pages will be printed.

The correction is a change to the logic in the \shipout macro, given below (which includes the change to the macro \loop given on p. 184 of the same issue). Each time TEX is to ship out a page, it tests to see if the page number is greater than the requested page. If so, it gets another number from the pages.tex file, continuing this test until either (a) the end of pages.tex is encountered, or (b) the requested page number is greater than or equal to the current page number.

```
\def\loop#1\repeat{\def\iterate
    {#1\expandafter\iterate\fi}%
    \iterate}
\def\breakout{\let\iterate\relax}
%    This lets us "escape" from a \loop
\let\repeat\fi
\def\shipout{\begingroup
    \loop
        \ifeof\pages \aftergroup\Shipout
        \else \ifnum\pageno>\nextpage
                \getnextpage
            \else \breakout
                \ifnum\pageno=\nextpage
                    \aftergroup\Shipout
                \else \aftergroup\Tosspage \fi
        \fi \repeat\endgroup }
```

(Note to users of the new \loop: I used \aftergroup to "hold off" on executing either \Shipout or \Tosspage because the \expandafter technique doesn't quite work due to the nesting of \if...\fi within \loop...\repeat. The alternative,

```
\expandafter\Tosspage\expandafter ...
```