

# **FWEB in Questions and Answers**

assembled for LitProg <litprog@shsu.edu>  
by Marcus Speh <marcus@x4u.desy.de>  
last updated 5 Jan 1993

This article contains frequently asked questions and their answers concerning the FWEB system of *Literate Programming* from the *LitProg* mailing list, until January 1993.

Most of the answers are taken from the postings, with little moderation from my side. The original postings are archived at `niord.shsu.edu` in directory `[FILESERV.LITPROG]`.

Comments, corrections and suggestions from several people were used to improve single questions and answers (see [Acknowledgements], page 5). Most notably, John Krommes, FWEB's designer, provided a wealth of useful comments. I have dubbed him *JAK* in some places.

Inside quotes, my own comments are marked [*. . . -MS*].

Sections of the FWEB User's Manual are referred to by their section numbers, as in “[M-10.2]” (see [Question 5], page 4).

# 1 Introduction and Editorial

## [1] What is FWEB?

For a thorough discussion of the WEB system, I refer to the extensive literature, collected in Nelson Beebe's bibliography (at `ftp.math.utah.edu` in directory `pub/tex/bib`). For pointers, see the upcoming *General FAQ* by Dave Thompson.

### The WEB system.

In D.E. Knuth's [author of the original WEB, coauthor of CWEB] own words, [M-2.4]

“The philosophy behind WEB is that an experienced system programmer, who wants to provide the best possible documentation of his or her software products, needs two things simultaneously: a language like  $\text{\TeX}$  for formatting, and a language like C for programming. Neither type of language can provide the best documentation by itself; but when both are appropriately combined, we obtain a system that is much more useful than either language separately.

The structure of a software program may be thought of as a *web* that is made up of many interconnected pieces. To document such a program we want to explain each individual part of the web and how it relates to its neighbours. The typographic tools provided by  $\text{\TeX}$ <sup>1</sup> give us an opportunity to explain the local structure of each part by making that structure visible, and the programming tools provided by languages such as C or Fortran make it possible for us to specify the algorithms formally and unambiguously. By combining the two, we can develop a style of programming that maximizes our ability to perceive the structure of a complex piece of software, and at the same time the documented programs can be mechanically translated into a working software system that matches the documentation.”

### FWEB - a multilingual WEB variant.

FWEB which is being supported and upgraded by John Krommes is a substantial extension of CWEB, FWEB supports C, C++, Fortran, Fortran90, Ratfor, and  $\text{\TeX}$  i.e. writing  $\text{\TeX}$  macros (style files). It runs on most platforms: VMS, PC, UNIX, and pretty much anything that the GNU C compiler (GCC) is supported for.

JAK [M-1]:

“The principal design contributions to this version of WEB are

1. the concept of a current language, so that one can process code written in multiple languages in the same WEB run;
2. new production rules for Fortran, Ratfor, and  $\text{\TeX}$  (and some modifications of Levy's<sup>2</sup> rules for C;
3. a C-like built-in macro preprocessor;
4. the ability to directly translate Ratfor into Fortran.

In addition, many miscellaneous details have been changed and a variety of convenience features has been added.”

---

<sup>1</sup> [which itself is written in Knuth's original Pascal WEB -MS]

<sup>2</sup> [Levy and Knuth are the authors of the C variant of WEB -MS]

## [2] How can I subscribe to LitProg?

If you are not yet subscribed to LitProg, but you are interested, send a MAIL message to 'LISTSERV@SHSU.BITNET, or `listserv@shsu.edu`, or `LitProg-Request@shsu.edu`' stating in the text of the MAIL: 'SUBSCRIBE LITPROG your name in quotes'. The list itself is unmoderated: messages sent to '`litprog@shsu.edu`' are automatically distributed to all subscribers.

## [3] How can I get the FWEB FAQ?

You may obtain the latest version of this FAQ via anonymous FTP from

- 'niord.shsu.edu' [192.92.115.8] in directory [FILESERV.FWEB],
- `lyman.ppp1.gov` [192.55.106.129] in directory `/pub/fweb/faq`,
- '`ftp.imada.ou.dk`' [129.142.128.14] in directory `/pub/faq`, and '`ftp.uni-stuttgart.de`' [130.83.55.75] in directory `pub/soft/tex/web/fweb`.

To retrieve the plain text file via e-mail, include

```
SENDME FAQ.FWEB
```

in the body of a mail message to

```
FILESERV@SHSU.BITNET (fileserv@shsu.edu).
```

Send the command

```
SENDME FILELIST
```

in a mail message to 'FILESERV' to get valid information on the names of the downloaded files if you'd like to have another version of the fweb-faq (see [Question 4], page 3).

NOTE: do **not** include the message 'SENDME FWEB' unless you want the full FWEB distribution!

## [4] Which versions of the FAQ are available?

Besides the text file `fweb-faq`, the FAQ is available as a Texinfo, a PostScript, an Info or a DVI file.

All these can be generated from the Texinfo source file `fweb-faq.texi`. The preprocessed DVI and PostScript files are not included anymore. See the simple makefile which accompanies the fweb-faq distribution for detailed information on how to process the Texinfo source. Short instructions are given in the file `README.fweb-faq`.

If you do not know anything about Texinfo, don't worry: With the `texinfo.tex` macro package, the principal source file `fweb-faq.texi` can be  $\TeX$ ed<sup>3</sup> like any ordinary  $\TeX$  file at the expense of not getting an index – try it!

If you would like to start with Texinfo, retrieve its latest version from '`prep.ai.mit.edu`' in directory `pub/gnu` (see [Question 6], page 4).

---

3

If you do not have  $\TeX$ , you may GNU's `texi2roff` for formatting.

## [5] What is the difference between the FAQ and the manual?

Most of the stuff in FAQs is in the manual, but people either don't read the manual, or it is too hidden to be found (the FWEB User's Manual currently has more than 200 pages). Thus the FAQ provides, if you will, an index to the manual for those questions that come up regularly. Where possible, sections of the User's Manual for the last release (FWEB version 1.23a) are referred to by their section numbers in square brackets, as in "[M-10.2]".

## [6] How can I contribute?

If you think that structure or content of this part of the FAQ can be improved, or if you think you have discovered an error, either write to me or post your comment to LitProg (with 'fweb-faq (version)' somewhere in the 'Subject' line).

If you find your name and email address anywhere on these pages, I'd be grateful if you check whether I got them right and inform me otherwise. The same for errors in ftp site addresses and Archive-Date (look into the Makefile to learn how to make these visible).

The FAQ for Literate Programming will probably not exist before there is a unified tool rather than a family of different environments. On LitProg, all these environments are subject to discussion, and a family of FAQ lists is planned, together with a master FAQ list for general questions. This general FAQ will be maintained by David B. Thompson <wqdbt@ttacs1.ttu.edu>.

If you are interested in participating in one of the FAQ lists for LitProg, either contact George D. Greenwade (E-mail: 'bed\_gdg@shsu.edu' [Internet], 'bed\_gdg@shsu' [Bitnet], 'shsu::bed\_gdg' [DECnet]), who maintains LitProg as well as the 'FILESERV' repository at Sam Houston State University, or simply announce your wish on the LitProg mailing list.

If you want to write a FAQ yourself, get "FAQ writer's FAQ" (by Nathan Torkington and Ian Kluft) which originates from the faq-maintainers mailing list. It can also be obtained via email by including 'SENDME FAQ.WRITING' in the body of a mail message to 'FILESERV', or retrieved from 'niord.shsu.edu' in [FILESERV.FAQ]FAQ.WRITING.

If you want to write your FAQ list in Texinfo, use `fweb-faq.texi` as a template for the principal source file (see [Question 4], page 3).

## [7] What else should be included in the FAQ?

- Merge the FAQs from the FWEB User's Manual with this list.
- Remove items of more general character in favor of Dave Thompson's general FAQ for LitProg.
- Give more details on the form of the FAQ file's Texinfo source?
- Add thoughts on OOP using the FWEB system (see [Question 16], page 8)?
- Should I include information on how to use FTP?
- Should this or another member of the LitProg FAQ list family be cross-posted to USENET (`comp.text.tex`, `news.answers`)?

Please send me your opinion on these items and suggest more things which should be done!

## Acknowledgements

Thanks to all those whose postings in LitProg I have used and who have sent me corrections, especially Bart Childs, Johannes Muller, David Thompson and Don Petcher for this number of the FAQ, and to Robert Chassell [FSF, GNU Texinfo and Elisp support] for help with some Texinfo bugs.

Thanks to the sysadmins who put up the files on FTP sites, especially Thorbjoern Ravn Andersen ([ftp.imada.ou.dk](ftp://imada.ou.dk)) and Joachim Schrod ([ftp.th-darmstadt.de](ftp://th-darmstadt.de)).

Special thanks to George D. Greenwade for maintaining the list and the file archives at Sam Houston University, and to John Krommes for checking this FAQ list for his child.

Personally, I feel indebted to Thorsten Ohl for having introduced me to FWEB in the first place, and having taught me many more things related to computing.

## 2 FWEB sources/Manual/Installation

### [8] Where can I get FWEB?

You can get FWEB (version 1.23a) via anonymous FTP either from `lyman.ppp1.gov`, in directory `pub/fweb/v1.23`:

```
/anonymous@lyman.ppp1.gov:/pub/fweb/v1.23:
```

```
-rwxr-xr-x  1 4145      589          8396 Apr 14 22:45 INSTALL.FWEB*
-rw-r--r--  1 4145      589        18967 Apr 14 22:45 Makefile
-rw-r--r--  1 4145      589         3068 Apr 14 22:45 READ_ME.FWEB
-rw-r--r--  1 4145      589           94 Apr 14 22:45 TAR.exclude
-rw-r--r--  1 4145      589       1419071 Apr 14 22:45 TAR.v1.23.Z
drwxr-sr-x  2 4145      589           512 Apr 14 22:45 demos/
drwxr-sr-x  2 4145      589         1536 Apr 15 18:06 manual/
...
```

Or from `niord.shsu.edu`, in directory `[FILESERV.FWEB]`:

```
/anonymous@niord.shsu.edu:/ANON_DEV:/FILESERV/FWEB:
```

```
FWEB-V1_23.TAR_Z;1      2772  21-JUL-1992 12:28 (RWE,RWED,RE,RE)
FWEB-V1_23.ZIP;1       2370  21-JUL-1992 13:10 (RWE,RWED,RE,RE)
...
INTRO.PS;1             175   22-JUL-1992 16:52 (RWE,RWED,RE,RE)
...
```

The PostScript file `INTRO.PS` gives a short introduction to Literate Programming in general, combined with a simple Fortran example program in FWEB.

### [9] What is the latest version of FWEB?

The latest version of FWEB is 1.23a - some time in January 1993, version 1.30-beta is bound to come out.

Versions that are explicitly intended to be beta versions are now indicated as such—e.g., `v1.30-beta`—in the banner line from the processors.

JAK: “However, given the fragmented state of my time, in some sense they’re all beta.”

The latest version can be retrieved via anonymous FTP as described in [Question 8], page 6.

### [10] Where can I get FWEB for the PC?

The official FWEB PC executables are generated as part of each FWEB release. Retrieve them from the FWEB distribution site ‘`lyman.ppp1.gov`’ (see [Question 8], page 6) buried deep down in ‘`pub/fweb/v1.23/boot/ibm/pc`’:

```
...
312 -rw-r--r-x  1 4145      589        304636 Apr 15 1992 ftangle.exe*
```

```

272 -rw-r--r-x  1 4145      589          268878 Apr 15  1992 ftangle0.exe*
280 -rw-r--r-x  1 4145      589          277032 Apr 15  1992 fweave.exe*
...

```

`ftangle0.exe` is `ftangle` without the Ratfor preprocessor (see [Question 26], page 14).

Hans-Hermann Bode <hhbode@dosuni1.bitnet> announced his precompiled files for the PC. They can be retrieved from

anonymous@niord.shsu.edu: [FILESERV.PC-WEB]:

```

OOINDEX.;1                2  23-JUL-1992 16:26:54.85
...
FWEBEXE.README;1          6  11-JUN-1992 00:15:18.00
FWEBEXE.READ_ME;1        42  9-JUN-1992 12:13:34.00
FWEBEXE.ZIP;1            1525 23-JUL-1992 16:06:33.26
FWEBEXE.ZIP-LST;1        2  23-JUL-1992 16:22:24.54
...

```

or in Europe from 'dione.rz.uni-osnabrueck.de' (in directory: 'pub/msdos/tex/web'), and from 'ftp.uni-stuttgart' (in directory 'soft/tex/web/cweb').

## [11] Is FWEB small enough to compile on a PC?

Yes. It is known to compile with MicroSoft C 6.00 and Turbo C.

JAK: "As a remark, it took some considerable pain to make this work. The larger sources, such as `ftangle.web`, had to be split into as many as 3 parts (handled by WEB macros and module names). The make file compiles each of those parts separately, then combines them at link stage. Before this was done, compilers tended to bomb with messages such as `Out of heap space`."

On LitProg, Tero Laakkonen <laakkonen@cc.helsinki.fi> reported that FWEB also compiles under linux-096c with gcc-2.2.2 on a 80386.

In general, John Krommes offers support for porting FWEB to new systems - contact <krommes@lyman.pppl.gov>.

## [12] Has anyone installed FWEB on an HP machine?

Either get GCC (the GNU C Compiler), or use (if you have it) the unbundled Hewlett-Packard ANSI C Compiler, which is invoked with `cc -Aa -D_HPUX_SOURCE` (as 'cc(1)' explains in detail). Then `make` should proceed without any problems.



## 3 Common Things People Want To Know

### [13] Can I use LaTeX with FWEB?

In principle, you may use any package of T<sub>E</sub>X macros you like. However, an arbitrary macro package may contain conflicts with macros used in `fwebmac.sty`. Such conflicts are supposed to be eliminated for LaTeX.

As for LaTeX which is probably the package most widely used, you have to do two things at minimum [M-19.7.22]:

1. Use the command-line option `-PL` (or place them into your initialization file `~/fweb` (see [Question 25], page 14).
2. Process fweave's output with `latex` instead of `tex`.

In most cases, to use LaTeX with FWEB it suffices to just say `'latex test'` instead of `'tex test'`. (I.e., an attempt is made to hide internally whatever differences there are.) It never hurts, and sometimes helps, to use the command-line option `-PL` (select post-processor LaTeX). To print a | in a `verbatim` environment, use `@|`, like: `'\begin{verbatim} ... @| ... \end{verbatim}'`.

Since FWEB overrides the `'\output'` routine of LaTeX, some clever page layout facilities will not work - e.g. there are no floating bodies: while the `'table'` environment is lost, the `'tabular'` environment still works. See [M-19.22] for details.

Bart Childs reports that the use of AMST<sub>E</sub>X causes some problems but that it has been done.

Difficulties with using LaTeX should be reported directly to Krommes.

For bringing in `'lex'` and `'yacc'` scripts and the problem of using LaTeX's `'\footnote'`, See [Question 14], page 8.

### [14] Can I use lex and yacc scripts with FWEB?

Lewis Perin <m-lp9966@jed.cs.nyu.edu> reported that he has succeeded bringing in `'lex'` and `'yacc'` UNIX scripts in for FWEB using LaTeX's `'verbatim'` environment.

### [15] Can I use the footnote environment from LaTeX?

Lewis Perin reports that the superscripts of `'\footnotes'` appear in the main text while the actual footnote is nowhere to be seen. In his reply, Zdenek Wagner <wagner@csearn.bitnet@shsu.edu> refers to a style file `ftn.sty` which he wrote to correct this (originally for use with CWEB, it seems).

`ftn.sty` can be retrieved from `niord.shsu.edu` [192.92.115.8] in directory `[FILESERV.STY]`.

### [16] Which WEB shall I use for C++ ?

As CWEB's coauthor Silvio Levy said, starting from version 2.9 (beta), CWEB will understand C++ syntax as well.

Successful compilation on the PC with Borland-C++ was reported on LitProg.

In the opinion of many people, FWEB is the best CWEB available. It also supports C, C++, F90, Ratfor, Ratfor-90, and writing T<sub>E</sub>X macros wherein (f)tangle produces .sty files.

As for the more general question of Object-Oriented Literate Programming, a very interesting discussion was started by Paul Lyon in December.

Hopefully, this topic will make it to Dave Thompson's General FAQ for LitProg (see [Question 6], page 4).

### [17] Does FWEB support Makefiles?

Not yet. This is planned, though [M-10.2].

### [18] FWEB with the GNU Emacs editor?

If you are developing your FWEB programs using the GNU Emacs editor, there is `web-mode.el` by Mark Motl <motl@cs.tamu.edu>; the corresponding GNU Emacs *mode* can deal with WEB, CWEB and FWEB.

It is capable of many things, including jumping to sections and modules, inserting (and previewing) index entries, hiding and exhibiting the body of a .web file (showing the tree), inserting, quoting, and consistently renaming modules etc. It supports change files and journal files. It is especially useful when dealing with large .web files not to have to deal with *monolithic* FWEB files.

For more information, you may obtain a 30pp. User's Manual and the source files from the author, from Bart Childs <bart@cs.tamu.edu>.

The current version of `web-mode` (v 1.61, as of Dec 28, 1992) is faster, more robust, and the documentation in the manual is improved and more accurate now.

You may also obtain this version as a 220 KByte shell archive from me <marcus@x4u.desy.de>.

### [19] How do I turn off double sided pagination in FWEB?

In FWEB, the command `\identicalpageheadstrue` [to be put in the so called *limbo* part of your FWEB file] makes all page headers identical, but it puts the page numbers in the upper LEFT corner and the section numbers in the upper RIGHT corner. As Don Petcher pointed out, to do it the other way around you can either change the macro '`\normaloutput`' in `fwebmac.sty` as indicated:

```
\def\normaloutput#1#2#3{\shipout\vbox{
  \ifodd\pageno\hoffset=\pageshift\fi
  \vbox to\fullpageheight
  {
    \iftitle\global\titelfalse
    \else\hbox to\pagewidth
      {\vbox to10pt{}}%
      \ifidenticalpageheads#3\else % THIS WAS ORIGINALLY #2 <-----■
        \ifodd\pageno
          #3% Makes page numbers alternate left \& right.
        \else#2\fi
      \fi
    }
  }
}
```

```

        \fi
        }%
\fi
\vfll#1
}}% Parameter |#1| is the page itself
\global\advance\pageno by1}

```

or you can include the above version in your T<sub>E</sub>X file as a redefinition of the macro.

## [20] Can I define (and format) new reserved words?

Assume you want to declare ‘far’ to indicate a function pointer:

```
void far (*reset)();
```

In order for fweave to treat ‘far’ like a reserved word in C (or any of the language supported by FWEB), say

```
@f far int
```

somewhere in your source. Weave does not by default recognize ‘|far|’ as a reserved word (this property extends to cweave as well).

The formatting with ‘@f’ is **language-specific**; it only applies to identifiers used in the language in force at the point the format statement is encountered. This feature allows a WEB programmer to invent new reserved words and/or to unreserve some reserved identifiers [M-11.12].

## [21] Symbolic debugging of FWEB files?

FWEB inserts sync lines ‘#line 137 foo.web’ into the code, so any compiler/debugger worth its money should respect them. (Including those running under MS-DOS.)

Extra FWEB comments in the tangled output can be suppressed by a simple command line switch. See [M-14.2] for a list of command line options.

NB: well thought out code needs less time debugging - comments like "since I've started using anyWEB, my debugger has just been collecting dust . . ." were reported from many people on the list.

## [22] Inserting meta-comments in FWEB files?

Johannes Muller reported strange output from fweave when defining two macros *debug* and *gubed* to enclose optional code for debugging purposes. As his global language, he chose C++.

```

@m debug @(
@m gubed @)
@f debug do
@f gubed while

```

This worked fine for him using WEB with Pascal. ‘@(' and ‘@)’ are control codes which mark the begin and end of a *meta-comment*, i.e. commented out code that will appear in the output file [M-11.38]. Though I wasn’t able to reproduce his errors, it should be remarked that the preferred way is to use the WEB preprocessor construction ‘@#if(0) . . . @#endif’ instead.

JAK: “That Muller’s example doesn’t work probably points out a problem with the macro preprocessor”.

For debugging purposes, one can bracket pieces of code by ‘`@#ifdef DEBUG ... @#endif`’, switching on *DEBUG*

by the commandline flag ‘`-mDEBUG`’ for ftangle [M-7.5].

## [23] Automatic referencing in documentation sections?

To produce the woven output

```
For info on porting, see section 5
```

As Steve Avery reported, you can cheat and, somewhere after you start the module, just throw in something like ‘`\let\refreporting=\modno`’ and then reference it by ‘`see section~\refreporting`’.

In FWEB, ‘`\modno`’ is set to the number of the module being typeset.

Another way to refer to a section is described in [M-19.7.18]:

```
@ Porting.
\modlabel{PORTING}
```

```
... more ...
```

```
@ Another section.
\modlabel{ANOTHER}
```

```
For info on Porting, see \WEBsection{PORTING}.
```

‘`section`’ is inserted automagically. In the same fashion, the label ‘`ANOTHER`’ allows the user to refer to that section number.

In LaTeX, forward referencing works, in Plain TeX it doesn’t. Unless you’re using LaTeX, the latter recipe requires that ‘`\modlabel{ANOTHER}`’ is defined before referring to it.

If you want to say *module* instead of *section*, use ‘`\WEBmodule`’ (or equivalently ‘`\module`’) instead of ‘`\WEBsection`’.

## [24] How do I make a title appear on the contents page?

Bart Childs <bart@cs.tamu.edu> contributed three short files to solve that problem. These listings will disappear here once the full `web-mode.el` distribution (see [Question 18], page 9) is available at ‘`FILESERV`’:

- `limbo.material` may be used as the initial skeleton for all WEB files. There is a line containing ‘`\def\title`’ which should be modified to include the ‘`title`’. Also note that the first few lines are to encourage a little more documentation about the source. A few lines further into this is a similar convenience to add an abstract that will appear on the cover page too.
- `limbo.sty` is a convenient place to record macros that are commonly used in WEB files.
- `time.tex`, required by `limbo.material`.

Note that the `limbo` style-file parameter can be used to automatically insert common material into the `limbo` section of an FWEB file.

### `limbo.material`

```

----- limbo.material -----
%
% ??????.web,   ?fweb version 1.23
% Author
% Address
% e-mail and phone
%
%   LIMBO MATERIAL
%
\input limbo.sty

\def\title{{\tt }}

%% Comment the previous and uncomment this if you don't use web-mode
%%\def\title{{\tt ?? I need a Title ??}}

%% begin Bottom of Contents Page macro
\def\botofcontents{\vskip 0pt plus 1fil minus 1.5in
{\bigskip\parskip6pt plus2pt \parindent20pt
%% begin abstract
\vskip0.5in
\noindent{\bf Abstract. }\it

}%% end abstract

%% BC often puts this in as a comment about pre-release versions ...

%\vskip0.5in
%{\vfill\it %% comments on anything else ????
%
%\vfil}% end of comments on anything else
\vfil
  \rightline{My Name}% You can personalize your output here, for example.
  \rightline{\today   }% today.tex should be preloaded, input it if not
  \rightline{\miltime }% time.tex  should be preloaded, input it if not

}%      end of botofcontents

%   END OF LIMBO MATERIAL
%
%
%   BEGINNING OF WEB

```

```

%

%% Delete the next line after appropriate substitution.
%% In fweb's you want an @c, @c++, @n, @n9, or @Lx at this point

@* First Module.

@* Index.

```

## limbo.sty

```

----- limbo.sty -----
% \input today %%% How Ridiculous, preload it!!!!!!!
\font\ninett=cmtt9
\font\ninerm=cmr9
\let\mc=\ninerm % medium caps for names like UNIX
\font\Csc=cmcsc10 % Computer Modern Computer Small Caps
\def\PASCAL{{\rm Pascal}}% Does very little
\def\WEB{{\ninett WEB}}% use like \WEB{} to make space significant
\let\web=\WEB \def\FWEB{{\ninett FWEB}} \let\fweb=\FWEB
\def\Fortran{{\Csc Fortran}} \def\Cee{{\bf C}}
\def\Unix{{\mc UNIX}} \let\unix=\Unix
\def\BSl{{\rm\char'134}} \def\<{\$\rangle\,$} \def\>{\$, \rangle$}
%
%% begin Top of Contents Page macro
%
\def\topofcontents{\hspace 6in
  \vspace -30pt plus 1fil minus 1.5in
  \centerline{\title}
  \vspace 15pt
  \centerline{\today}
\bigskip \vfill
  \def?##1]{\hbox to 1in{\hfil##1.\ }}% end of topofcontents

```

## time.tex

```

----- time.tex -----
\newcount\milhours\newcount\minutes\newcount\hours
\def\thetime{\milhours=\time
\divide\milhours by 60
\minutes=\milhours
\multiply \minutes by -60
\advance\minutes by \time
\hours=\time \divide\hours by 60
\ifnum \hours>12 \advance \hours by -12 \fi
\the\hours:\ifnum \minutes > 9 \the\minutes \else 0\the\minutes \fi}

\def\miltime{\milhours=\time

```

```

\divide\milhours by 60
\minutes=\milhours
\multiply \minutes by -60
\advance\minutes by \time
\the\milhours:\ifnum\minutes>9 \the\minutes\else 0\the\minutes \fi}

```

## [25] Key differences for FWEB on different systems?

The machine dependent files, especially `custom.h` and `defaults.mk` necessary for bootstrapping, can be found in the current distribution of FWEB (v1.23a) in subdirectory `/boot/`. See the `READ_ME.*` files there for details.

To run FWEB, commonly used options can be put into an initialization file [M-14.3] (default name `.fweb`). The subdirs of `/boot/` contain sample `.fweb` files with necessary commandline options for the system on which FWEB was built.

## [26] Why is FWEB so huge?

Most of FWEB's size can be attributed to the need of supporting vastly different languages. The input routines for a free form language are not usable for Fortran and vice versa. Furthermore, FWEB has a Ratfor processor built in (which is a **big** plus, though it is not strictly related to literate programming).

JAK: "Two other reasons for the large size are

1. the built-in macro preprocessor, an extended version of that for ANSI C;
2. the style-file mechanism.

On the positive side, FWEB does most of its memory allocations dynamically, so one can cut down the size of various tables if necessary. Type `'ftangle -Y'` to find out about the current allocations; use the `-y` option to change them. The statistics option `-s` is also helpful to find out how much a job actually uses."

If you do not want the Ratfor preprocessor, you can make a smaller `ftangle` with the option `'LOAD\_RATFOR=0'` (see the file `web/ratfor0.web` of the FWEB distribution, and [5]).

## [27] What is the difference between CWEB and FWEB?

See [M-19.8], Appendix H, for a short list comparing these two variants of the WEB system.

## [28] What is the difference between FWEB and Funnelweb?

The following was extracted from a text by Paul Lyon (and will eventually be merged with a FAQ for Funnelweb . . .):

"FWEB and Funnelweb are quite distinct. FWEB is built on top of the CWEB framework; although the parser in its weave processor can do more—all of ratfor, C, C++, and (though the support is not complete),  $\TeX$ , it is still confined to those specific languages, and still imposes on the user the formatting conventions that please its author. To get something different you will have to hack the weave processor. (One could possibly get somewhere by modifying the  $\TeX$  macro package that FWEB uses, but that might be the harder way to go, unless, of course, you are already a  $\TeX$ pert) . . .

Funnelweb, on the other hand, does not try to parse the *source* code at all; it just takes the layout of the source as written, turns off the meaning of plain  $\TeX$  special characters, sets typewriter font, and then invokes ‘`\obeylines`’ and ‘`\obeyspaces`’; all this together causes  $\TeX$  to print the source verbatim (the paragraph formatting is turned off, and  $\TeX$  does not gobble spaces). The original WEB, CWEB, FWEB, and Spiderweb all parse and format the source, inserting  $\TeX$  math codes for the operators, putting keywords in boldfont, adjusting the indentation, and so on. If you like the style chosen for the programme, it looks much nicer that way. Except for Spiderweb, which can be adapted to various languages by allowing a fair range of variation in specifying the pretty printing grammar using a large ‘`awk`’ script to process the grammar spec and generate replacement code for significant chunks of weave, the pretty printing parser(s) in the other are hard coded. This has it uses besides making the typeset code more attractive; WEB, CWEB, FWEB, and Spiderweb all do an index of identifiers for the code that can differentiate, for the most part, between declaration and use of an identifier (they know enough about the grammar to do that, but not, of course, as much as a compiler or interpreter) . . . ”

[Rest deleted: continues with more details on Funnelweb.]

JAK: “Don’t hack `fweave`! Many effects can be obtained by modifying `fwebmac.sty`. (If one knows enough to hack `fweave`, he presumably knows enough to hack `fwebmac.sty` instead.) Note that the style-file mechanism does provide some degree of customization.”

Please report customizations that are really necessary to John Krommes.

## [29] Do FWEB files necessarily have to be monolithic?

[Later, this item might move to the general FAQ by David B. Thompson]

There is no real support for multiple source files under FWEB. What some might like to see is a solution similar to the one presented by Cameron Smith in his KR-CWEB-SAMPLE distribution. He had to hack CWEB’s `cwebmac.sty` to get a neat table of contents and a combined index distinguishing between entries from different source files [in fact, ‘KR-CWEB-SAMPLE’ illustrates many more things, all more or less related to breaking up a literate C program into multiple source files]. For illustration, this is how the table of contents should look like (sample from Cameron’s files):

```

source module 1: main          Sect Page
introduction.....1.1  1.1
main  pgm.....1.4  1.2
index.....1.10  1.4

source module 2: getop
introduction.....2.1  2.1  etc.
```

while entries of the combined index are separated like

```

Index for source module 1: main
...
Index for source module 2: getop
...
...
```

```

Sections in source module 1: main
```



```
<case for newlines 9> used in section 4...
...
```

For FWEB, the inclusion of files using '@i' is a partial solution [M-11-13] to avoid having to put all the source code in one file. So far, nobody has presented a solution in Cameron's spirit.

This feature is put on JAK's list of possible future enhancements. It will not be in c1.30, though. Note that CWEB has got an @i option, too (although FWEB's is somewhat more general).

Yet another approach (not limited to FWEB) is the use of a *smart* editor like GNU Emacs, combined with a *smart* tool like `web-mode` which is effectively hiding many of the mischiefs of monolithic source files from the programmer. (see [Question 18], page 9).

### [30] How did tangle and weave get their names?

[Later, this item might move to the general FAQ by David B. Thompson]

In a reply, Cameron Smith wrote:

"One of the problems with having a single preeminent writer in a language is that everyone always assumes that any enduring tidbit came from his pen . . .".

Fortunately, for the question *How did tangle and weave get their names*, the circulus vitiosus of erroneously quoting William Shakespeare could be broken. The full answer:

```
O, what a tangled web we weave
When first we practise to deceive!
-- Sir Walter Scott, Marmion, canto 6, verse xvii (1808)
```

Cameron also suggested to include these lines by Richard Palais (1982):

```
O, what a tangled WEB we weave
When TeX we practise to conceive!
```

Concerning FWEB, JAK reluctantly remarks [M-1]: "We shall call this new version FWEB when necessary . . .".

And more recently: "I now think the choice of 'F' in FWEB was a mistake. 'F' stands for Fortran, which motivated this project, but the ability of FWEB to handle multiple languages is one of its most distinctive and useful features. But it's probably too late now."

The question to which extent *WEB* inherits from certain German words I'll leave to the native speakers (with a light heart).

### [31] How am I supposed to abbreviate "Literate Programming"?

[Later, this item might move to the general FAQ by David B. Thompson.]

There wasn't a clean vote on the mailing list. Most people seemed to agree that one should not misuse common abbreviations like *LP*, *LPR* etc. Also, the acronym should be pronounceable (thus eliminating *LitPgm* e.g.). The list's name *LitProg*, proposed by Cameron Smith, seems to be accepted by a majority now. This is also the acronym adopted for the FAQ list.

## 4 General Questions/Bugs/Problems

### [32] What if I think I found a bug in FWEB?

John Krommes <krommes@lyman.pppl.gov> actively supports FWEB – he must be considered the ultimate source of wisdom.

Once you think you've found a bug, put it out on the list. If nobody responds, cut your file down to the smallest subfile that still exhibits the problem you encountered. Then you may contact John.

### [33] Problem opening a new output file for tangled code?

If you're using FWEB's '@o' option in order to open a new output file for tangled code (with local scope - i.e. for the duration of the current section only [M-11.21]), you may see output looking like this:

```
> ftangle test
This is ScRaMbLeD FTANGLE [SunOS/UNIX version 1.23a (April 13, 1992)].
Reading test.web ... *1 *4 *6
Writing the output file(s): (test.c)(getop.hQmSWs)
Done.
CPU = 0.3 sec.; REAL = 0.8 sec. CPU/REAL = 45.0%.
[FTANGLE: No errors were found.]
```

with spurious characters in the output file names.

This is a bug which will be fixed in FWEB version 1.23b as promised in the README file of the FWEB distribution v1.23a.

### Preliminary Bug Report

Aside the list, Don Petcher reports a more specific version of the problem (this seems to affect both the local '@o' and the global scope command '@O'):

If you change output files with the '@O' command (or the '@o' command) and the second filename is shorter than the first, then the second filename gets spurious characters added.

E.g. creating \*.cc files and \*.h files, the \*.h file always gets an extra character after the first writing to the \*.cc file. A temporary cludge is obvious - add an extra character to the '\*' part of the \*.h file. So your file names don't go together well, but you escape the problem.

It now seems as if this bug is pretty much machine dependent. The recipe pointed out in the last edition of this list, adding characters to the name part of the secondary output file, only works for Don Petcher on his NeXT workstation.

Stirred up by Johannes' complaint Don and I tried a couple of silly programs on his NeXT, on a HP9000, on an Apollo 700, on a Sun-4 and on a Silicon Graphics workstation (SGI). The result was that none of the machines behaves the same for the programs we tried. The Apollo did not exhibit the bug at all, for the NeXT Don's cludge works. The HP accepts output file names "g" and "g." but produces spurious characters for "g.h" - the Sun only accepts "g" but not "g." and "g.h". The SGI exhibits the error if the filename contains more than one underscore or if the filename is longer than six letters.

JAK: “I think this is fixed. I will attempt to thoroughly test this out before releasing v1.30.”

### [34] Maximum number of equal characters in different section names?

FWEB may have problems distinguishing section names, if one of them is an exact subset of a longer name rather than a length.

This is a bug in the mind of some, but it may as well be regarded a good *feature* as it should be a warning that poor choices of names has happened.

JAK: “I think it’s a feature, not a bug. And unless I screwed something up, the behavior derives from CWEB.”

### [35] Must I leave leading blanks in my fortran FWEB file?

Yes, you should. This is proven by the following example coming from Bart Childs.

```
@n
@* The world is not perfect.
This wonderfully short problem blows outputs garbage
from fweave and ftangle if you don't have the leading
six columns blank. Be careful, because in many cases
FWEB works fine without the requisite leading blanks.
```

```
@a
  program final_exam
  implicit none
  integer i, j
  do i = 1, 7, 2
    do j = 6, 1, -1
      write(*,*) i, j
    end do
  end do
end
```

To see the effect on fweave, indent with 4 leading blank spaces instead of 6.

JAK: “The input reader for Fortran-77 **follows the rules** of Fortran-77—i.e., statement labels in columns 1–5, continuation character in column 6, columns 6–72 for code. However, note that Ratfor uses free-form syntax, and Fortran-90 also has a free-form syntax mode, which FWEB attempts to support.”

# Index

## #

#line ..... 10

—

-PL, select pre-processor LaTeX ..... 8

-s, FWEB statistics option ..... 14

•

.fweb ..... 8

/

/anonymous@lyman.pppl.gov/pub/fweb/v1.23... 6

/anonymous@niord.shsu.edu/FILESERV/FWEB... 6

[

[M-1] ..... 2

[M-10.2] ..... 9

[M-11.12] ..... 10

[M-11.21] ..... 17

[M-11.38] ..... 10

[M-14.2] ..... 10

[M-19.7.18] ..... 11

[M-19.7.22] ..... 8

[M-19.8] ..... 14

[M-2.4] ..... 2

[M-7.5] ..... 10

@

@( ..... 10

@) ..... 10

@|, in LaTeX verbatim ..... 8

@f ..... 10

@o ..... 17

@O ..... 17

\

\footnote, LaTeX ..... 8

\identicalpageheadstrue ..... 9

\modlabel ..... 11

\output ..... 8

\ref ..... 11

\WEBsection ..... 11

|

|, in LaTeX verbatim ..... 8

## A

AMSTeX ..... 8

anonymous FTP ..... 3, 6

anonymous@dione.rz.uni-osnabrueck.de ..... 6

anonymous@ftp.imada.ou.dk/pub/faq ..... 3

anonymous@ftp.uni-stuttgart.de ..... 6

anonymous@ftp.uni-stuttgart.de/pub/soft/tex/web/fweb... 3

anonymous@lyman.pppl.gov ..... 6

anonymous@niord.shsu.edu/[FILESERV.PC-WEB]... 6

anonymous@niord.shsu.edu/FILESERV/FWEB... 3

ANSI C ..... 7

Appendix H ..... 14

Archive-Date ..... 4

## B

Bart Childs <bart@cs.tamu.edu> .... 5, 8, 9, 11, 18

Bugs ..... 17

## C

C ..... 2

C++ ..... 2, 8

Cameron Smith

<cameron@symcom.math.uiuc.edu> ..... 15, 16

Change files ..... 9

Command line options ..... 10

Contents page ..... 11

CWEB ..... 2, 8

CWEB, @i option in ..... 16

## D

D.E. Knuth (DEK) ..... 2

David Thompson <wqdbt@ttacs1.ttu.edu> .... 2, 5

Debugger ..... 10

Debugging ..... 10

DEK (Donald E. Knuth) ..... 2

Design contributions ..... 2

Distinguishing section names ..... 18

Don Petcher

<petcher@wuphys.wustl.edu> ..... 5, 9, 17

Double sided pagination ..... 9

**F**

FAQ for FWEB	4
FAQ Writer's Guide	4
FAQ, General	4
FILESERV	4
Floating bodies	8
Footnotes in LaTeX	8
Formatting	2
Fortran	2
Fortran-77	18
Fortran-90	18
Fortran90	2
Ftangle	17
ftangle, change allocation	14
ftangle, current allocation	14
ftn.sty	8
Function pointer	10
fweave	15
fweb-faq	4
fweb-faq.dvi	3
fweb-faq.info	3
fweb-faq.ps	3
fweb-faq.texi	3
FWEB executables	14
FWEB for the PC	6
FWEB for the PC, precompiled files	6
FWEB sources	6
FWEB User's Manual	1
FWEB User's Manual, quoting	4
FWEB, future enhancements	16
FWEB, latest version	6
FWEB, painful work with	7
FWEB, v1.30-beta	6
fwebmac.sty	9, 15

**G**

gcc-2.2.2	7
GCC	7
General Questions	17
George D. Greenwade <bed_gdg@shsu.edu>	4
Global scope	17
GNU C compiler (GCC)	2
GNU Emacs editor	3, 9

**H**

Hans-Hermann Bode <hhbode@dosunil.bitnet>	6
Hewlett-Packard	7
How can I contribute?	4
HP	7

**I**

Ian Klufft <ikluft@uts.amdahl.com>	4
Initialization file	8
Intel 80386	7
Introduction and Editorial	2

**J**

JAK (John A. Krommes)	1, 16
JAK, quoted	2, 6, 7, 10, 15, 17, 18
Joachim Schrod <schrod@iti.informatik.th-darmstadt.de>	5
Johannes Muller <jmu@ccr.jussieu.fr>	5, 10
John A. Krommes (JAK)	1
John Krommes <krommes@lyman.pppl.gov>	2, 5, 7, 17
Journal files	9

**L**

Language-specific	10
LaTeX	8
Lewis Perin <m-lp9966@jed.cs.nyu.edu>	8
lex	8
limbo, style-file parameter	12
limbo.material	11, 12
limbo.sty	11, 13
linux-096c	7
Literate Programming, abbreviation for	16
LitProg Mailing List	1
LitProg-Request@shsu.edu	3
Local scope	17

**M**

Macro preprocessor	2, 10
mail-server@rtfm.mit.edu	6
Make	7
Makefile	4, 9
Makefiles, FWEB support	9
Marcus Speh <marcus@x4u.desy.de>	1
Mark Motl <motl@cs.tamu.edu>	9
Marmion, canto 6, verse xvii	16
Meta-comment	10
Microsoft C 6.00	7
MS-DOS	10
Multilingual WEB variant	2

**N**

Nathan Torkington <Nathan.Torkington@vuw.ac.nz>	4
Native speaker	16
Nelson H. F. Beebe <beebe@math.utah.edu>	2
niord.shsu.edu/FILESERV/LITPROG	1

**O**

Object-Oriented Literate Programming	8
--------------------------------------	---

**P**

Paul Lyon <uunet!emx.cc.utexas.edu!plyon> . . . . 14  
 PC . . . . . 2  
 prep.ai.mit.edu:/pub/gnu . . . . . 3  
 Programming, Literate (!) . . . . . 2

**Q**

Question for the General FAQ . . . . . 2, 15, 16

**R**

Ratfor . . . . . 2, 18  
 Ratfor processor . . . . . 14  
 README.fweb-faq . . . . . 3  
 Referencing . . . . . 11  
 Reserved words . . . . . 10  
 Richard Palais . . . . . 16  
 Robert J. Chassell <bob@gnu.ai.mit.edu> . . . . . 5

**S**

Sam Houston State University . . . . . 4  
 Section names, referencing of . . . . . 11  
 Silvio Levy <levy@math.berkeley.edu> . . . . . 2, 8  
 Sir Walter Scott <not reachable via email> . . . . . 16  
 Steve Avery <stevea@vast.unsw.edu.au> . . . . . 11  
 SUBSCRIBE LITPROG . . . . . 3  
 support porting FWEB . . . . . 7  
 Sync lines . . . . . 10

**T**

table . . . . . 8  
 tabular . . . . . 8  
 Tangle . . . . . 8  
 Tero Laakkonen <laakkonen@cc.helsinki.fi> . . . . . 7  
 Texinfo system . . . . . 3  
 TeX . . . . . 2  
 Thorbjoern Ravn Andersen  
   <ravn@imada.ou.dk> . . . . . 5  
 Thorsten Ohl <ohl@chico.harvard.edu> . . . . . 5  
 time.tex . . . . . 11, 13  
 Title . . . . . 11  
 Titlepage . . . . . 11  
 To-do-list . . . . . 4  
 Turbo C . . . . . 7

**U**

UNIX . . . . . 2

**V**

version 1.23b . . . . . 17  
 VMS . . . . . 2

**W**

Weave . . . . . 10  
 web-mode.el . . . . . 9  
 WEB . . . . . 2  
 WEB preprocessor . . . . . 10  
 William Shakespeare . . . . . 16

**Y**

yacc . . . . . 8

**Z**

Zdenek Wagner  
   <wagner@csearn.bitnet@shsu.edu> . . . . . 8



## Short Contents

1	Introduction and Editorial . . . . .	2
2	FWEB sources/Manual/Installation . . . . .	6
3	Common Things People Want To Know . . . . .	8
4	General Questions/Bugs/Problems . . . . .	17
	Index . . . . .	19





# Table of Contents

<b>1</b>	<b>Introduction and Editorial</b>	<b>2</b>
[1]	What is FWEB?	2
[2]	How can I subscribe to LitProg?	3
[3]	How can I get the FWEB FAQ?	3
[4]	Which versions of the FAQ are available?	3
[5]	What is the difference between the FAQ and the manual?	4
[6]	How can I contribute?	4
[7]	What else should be included in the FAQ?	4
	Acknowledgements	5
<b>2</b>	<b>FWEB sources/Manual/Installation</b>	<b>6</b>
[8]	Where can I get FWEB?	6
[9]	What is the latest version of FWEB?	6
[10]	Where can I get FWEB for the PC?	6
[11]	Is FWEB small enough to compile on a PC?	7
[12]	Has anyone installed FWEB on an HP machine?	7
<b>3</b>	<b>Common Things People Want To Know</b>	<b>8</b>
[13]	Can I use LaTeX with FWEB?	8
[14]	Can I use lex and yacc scripts with FWEB?	8
[15]	Can I use the footnote environment from LaTeX?	8
[16]	Which WEB shall I use for C++ ?	8
[17]	Does FWEB support Makefiles?	9
[18]	FWEB with the GNU Emacs editor?	9
[19]	How do I turn off double sided pagination in FWEB?	9
[20]	Can I define (and format) new reserved words?	10
[21]	Symbolic debugging of FWEB files?	10
[22]	Inserting meta-comments in FWEB files?	10
[23]	Automatic referencing in documentation sections?	11
[24]	How do I make a title appear on the contents page?	11
	<code>limbo.material</code>	12
	<code>limbo.sty</code>	13
	<code>time.tex</code>	13
[25]	Key differences for FWEB on different systems?	14
[26]	Why is FWEB so huge?	14
[27]	What is the difference between CWEB and FWEB?	14
[28]	What is the difference between FWEB and Funnelweb?	14
[29]	Do FWEB files necessarily have to be monolithic?	15
[30]	How did tangle and weave get their names?	16
[31]	How am I supposed to abbreviate “Literate Programming”?	16
<b>4</b>	<b>General Questions/Bugs/Problems</b>	<b>17</b>
[32]	What if I think I found a bug in FWEB?	17

[33] Problem opening a new output file for tangled code?..... 17  
[34] Maximum number of equal characters in different section names? .. 18  
[35] Must I leave leading blanks in my fortran FWEB file?..... 18

**Index..... 19**