/* ——— *

src2tex* version 2.12 (Sep 9, 1996)

Kazuo Amano† and Shinichi Nomoto‡

Abstract

One of the authors has a strong desire for combining documentation and manual with source program by using TEX's beautiful text and PostScript figures without any big literate programming tools, such as WEB system ([8]) or something like that ([15]). Unfortunately, most of literate programming tools require too much laborious effort to learn their usages, and also, as far as the authors know, no such tools allow to patch PS and EPS figures upon text. It is quite convenient if source program itself is its document, manual and releasenote simultaneously, in which you can explain usages, programming techniques, algorithms, theoretical backgrounds, *etc* in terms of high quality TEXt, mathematical formulae and beautiful figures. Here the authors would like to release softwares **src2tex** and **src2latex**. Those softwares would give an easy-to-use unified environment of source program, documentation and manual. They believe that **src2tex** and **src2latex** work effectively for a relatively small set of source programs, such as education and experiment programs. In fact, their src2tex-project itself consists of several education programs, and **src2tex** has worked very well in their project. For practical usages of **src2tex** and **src2latex** it would be better to compare *newton.c*, *simpson.c*, *hanoi.c*, *etc* with *newton.c.ps*, *simpson.c.ps*, *hanoi.c.ps*, *etc* respectively.

1. Introduction

Our softwares **src2tex** and **src2latex** are a sort of text converters from various types of source program files to plainTEX and LATEX format files, *e.g.*,

`src1.c` ⇒ `src1.c.tex`, `src2.f` ⇒ `src2.f.tex`.

However, **src2tex** and **src2latex** are not simple pretty-printers. Our **src2tex** and **src2latex** are designed to fulfill the following desires:

(1) **src2tex** and **src2latex** can identify differences of various computer languages, such as BASIC, C, C++, OBJECTIVE-C, COBOL, FORTRAN, HTML, JAVA, LISP, MAKE, PASCAL, PERL, SCHEME, SHELL, TCL/TK; ASIR, MACSYMA, MAPLE, MATHEMATICA, MATLAB, MAXIMA, MuPAD, OCTAVE, REDUCE.**

(2) **src2tex** and **src2latex** allow to use TEX's powerful typesetting mechanism within comment area of source program.

---

* Permission to use, copy, and modify this software and its documentation is granted under no conditions. However, the authors would be very happy if users could inform any modifications to kamano@tansei.cc.u-tokyo.ac.jp. Since src2tex is a free software, there is no warranty of any kind for the program.

† Department of Mathematics, Josai University, Keyakidai, Sakado, Saitama, 350-02, JAPAN (kamano@po.iijnet.or.jp)

‡ Department of Mathematics, Josai University, Keyakidai, Sakado, Saitama, 350-02, JAPAN (m93093@crescent.josai.ac.jp)

** ASIR, JAVA, MACSYMA, MAPLE, MATHEMATICA, MATLAB, REDUCE are registered trademarks of Fujitsu Laboratories, Ltd., Sun Microsystems, Inc., Macsyma Inc., Waterloo Maple Software, Wolfram Research, Inc., The MathWorks, Inc., The Rand Corporation respectively.

(3) **src2tex** and **src2latex** enable to patch PS and EPS files upon source file without any difficulties.
(4) **src2tex** and **src2latex** accept EUC Kanji code characters.

Remark. Kanji : a Japanese system of writing based on the Chinese one and composed principally of characters borrowed or adapted from Chinese, a single character belonging to the kanji system of writing (quoted from "Webster's Third New International Dictionary of the English Language Unabridged", *G. & C. Herrian Company*, 1971)

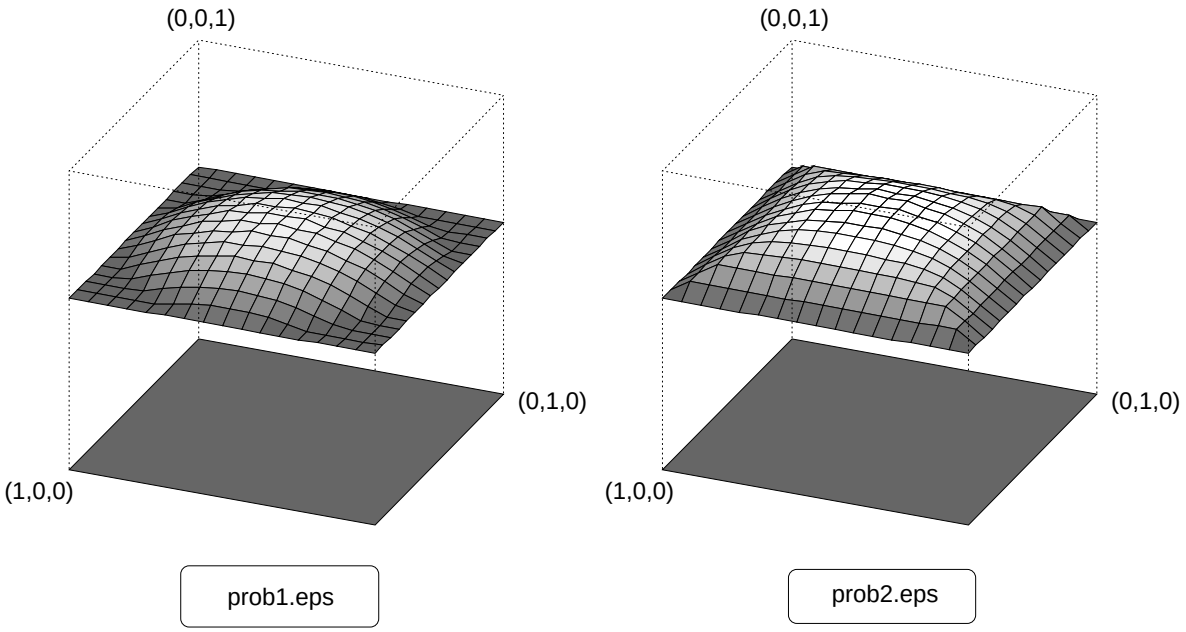
Using **src2tex** or **src2latex**, you can embed mathematical formulae into comment area of source program. For instance, you can insert formulae

$$\sum_{\nu=1}^{n} \nu = \frac{n(n+1)}{2}$$

$$\Gamma(x+1) = x! \sim \sqrt{2\pi x}\, x^x\, e^{-x}$$

$$\sqrt{\pi}\,(2n-1)!!\frac{(1-p)^n}{p^{n+\frac{1}{2}}} = \int_0^\infty e^{-pt}\,\frac{H_{2n}\left(\sqrt{t}\right)}{\sqrt{t}}\,dx$$

into comment area of C, like this. You can even include EPS files, by virtue of *dvi2ps* or something like that, as follows:



2. Compilation and Installation

In order to compile and install **src2tex** and **src2latex**, you have only to issue the following commands:

```
% make
% make install
```

After that, you would find executable files **src2tex** and **src2latex** in your current directory. You could copy them to a suitable place, *e.g.*,

```
% cp src2tex ~/bin/
```

```
% cp src2latex ~/bin/
```

If you could not make executables, it would be better to read PostScript files *fileio.c.ps*, *getdata.c.ps*, *langflag.c. ps*, *modflag.c.ps*, *pas_bold.c.ps*, *src2tex.c.ps*, *text2tex.c.ps* and *tools.c.ps*.

Remark 1. The authors are very glad if you can type

```
% make report
```

and send your local */etc/motd* file to kamano@po.iijnet.or.jp. They think this would give invaluable information for further development of **src2tex**.

Remark 2. If you are a DOS user, you have only to type

```
% make dos
```

You will get **src2tex.exe** and **src2ltex.exe** in the present working directory.

Furthermore, shell scripts **src2tex2dvi** and **src2tex2ps** might be useful. **Src2tex2dvi** and **src2tex2ps** are text converters from source program file to DVI and PostScript files. Those shell scripts are nothing more than combinations of free softwares *src2tex*, *jtex*, *dvi2ps*, *nkf* and *psnup*. Maybe, it is necessary to modify **src2tex2dvi** and **src2tex2ps** so that they work properly at your site (*cf.*, also, *src2latex2dvi* and *src2latex2ps*). For DOS users, we would like to provide two batch files *src2dvi.bat* and *tex2tex.bat* (*cf.*, *DOS_USER*).

3. Manual

The usage of **src2tex** and **src2latex** is quite simple. It suffices to input

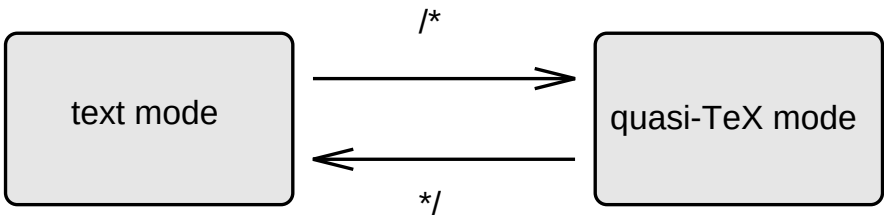
```
% src2tex source-file-name
% src2latex source-file-name
```

on the command line. No options are available and no options will be added, since we believe that it is no good to implement various options and circumvent programming difficulties. Their usage is so simple that we do not feel any necessities to write online manuals. If no input-file-name is given, **src2tex** and **src2latex** read data from *standard input* and write them out to *standard output*. When *source-file-name* is given, **src2tex** and **src2latex** try to identify language which is used in the given file. First, they see *file-name-suffix* and determine language type as follows :

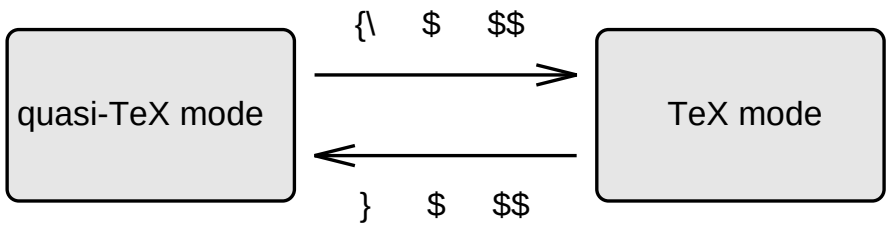
```
.tex, .txt            ⇒   TEXT
.bas, .vb             ⇒   BASIC
.c, .cpp, .vc         ⇒   C, C++, OBJECTIVE-C
.cbl, .cob            ⇒   COBOL
.f, .for              ⇒   FORTRAN
.html                 ⇒   HTML
.java                 ⇒   JAVA
.el, .lsp, .sc, .scm  ⇒   LISP, SCHEME
 makefile             ⇒   MAKE
.p, .pas, .tp         ⇒   PASCAL
.pl, .prl             ⇒   PERL
.sh, .csh, .ksh       ⇒   SHELL
.tcl, .tk             ⇒   TCL/TK
.asi, .asir, .asr     ⇒   ASIR
.mac, .max            ⇒   MACSYMA, MAXIMA
.map, .mpl            ⇒   MAPLE
.mat, .mma            ⇒   MATHEMATICA
.ml, .mtlb, .oct      ⇒   MATLAB, OCTAVE
.mu                   ⇒   MuPAD
.red, .rdc            ⇒   REDUCE
```

Next, they search key words and attempt to determine language. After that, if there exists a file *src2tex.s2t* [resp. *src2latex.s2t*], then **src2tex** [resp. **src2latex**] simply includes it at the beginning of output procedure. In case of DOS, *src2ltex.s2t* is read instead of *src2latex.s2t*.

Our **src2tex** and **src2latex** have three modes: text mode, quasi-TEX mode and TEX mode. First, you are in default text mode. As you can imagine, text mode is nothing more than typewriter mode. In text mode, "what you see is what you get". Second, you find that you are in quasi-TEX mode when you enter into comment area. For instance, if input file is C source file, then



In quasi-TEX mode of **src2tex**, you can input any characters as in text mode except the following four key words {\ , } , $ and $$. However, it is to be noted that font and spacing sizes are different from those of text mode. Third, when you are in quasi-TEX mode of **src2tex**, you are allowed to enter into TEX mode and escape from it by using one of the following pairs: {\ and } , $ and $, $$ and $$. To be more explicit,



Literally, TEX mode is genuine plain TEX mode. You can use plain TEX there without any restrictions. **Src2tex** is designed to use the above key words as follows:

$*mathematical formulae*$

$$*display style mathematical formulae*$$

{\ *TeXt and mathematical formulae*}

Remark 1. In the above expressions $, $$ and \ are all passed to TEX transparently. However, braces { and } are both replaced with blank spaces. For example, a phrase

{\bf bold face}

is translated into

\bf bold face

and passed to TEX. If you really want to get {\bf bold face}, then you have to use

{{\bf bold face}}

instead. Actually,

{{\ *TeXt and mathematical formulae*}}

is the other door to TEX mode. This method provides a safe encapslated transition from TEX mode to quasi-TEX mode.

Remark 2. **Src2tex** and **src2latex** have several fail-safe mechanisms. If you do not write mathematical formula properly, **src2tex** thinks that you have no knowledge of TEX, *i.e.*, you are not allowed to use TEX. For instance, if you really want to enter into TEX mode, it is better not to write

```
$T_abc$ .
```

As a substitute, you should use either

```
$T_{abc}$      or      $T_a bc$
```

(*cf.* [9]). In case of BASIC or PERL, `$` sign which means string variable is not always recognized as a TEX mode transition key word. For example, you cannot enter into TEX mode with

```
A$, str1$, ...
```

in BASIC comment area. You are not able to use TEX mode of PERL with

```
$A, $str1, ...
```

either. It would be better to use $$ or {\ in BASIC and PERL.

Remark 3. As you can easily imagine, in quasi-TEX mode of **src2latex**, you have to use \(, \) , \[and \] instead of $, $, $$ and $$ respectively. In case of **src2latex**, if you want to use a certain non-default documentstyle, say `twocolumn, 12pt, jarticle` style, then you have only to insert a comment area with a phrase

```
{\documentstyle[twocolumn,12pt]{jarticle}}
```

at the beginning of source file. The general form of this phrase is given by

{\documentstyle[*latex option*]{*latex style*}}

If you have already installed *dvi2ps* or something like that, you can include EPS files when you are in TEX mode. In case of *dvi2ps*, a line

{\special{epsfile=*eps file name ...*}}

would suffice to patch an EPS figure upon comment area.

For practical usages, it is better to read source files of **src2tex** (*e.g.*, fileio.c, getdata.c, langflag.c, modflag.c, src2tex.c, text2tex.c, tools.c), since they are written in src2tex style. For instance, if you input three commands

```
% src2tex src2tex.c
% tex src2tex.c.tex
% dvi2ps src2tex.c.dvi > src2tex.c.ps
```

in the source files directory of **src2tex**, you will get a PostScript file

```
src2tex.c.ps
```

which would show most of useful technicalities of **src2tex**. We would like to suggest you to compare *src2tex.c* with *src2tex.c.ps* carefully by using one of PostScript pre-viewers. You could learn practical usages quite easily.

4. Dirty Tricks

There is a dirty trick which enables to change certain global variables of **src2tex** and **src2latex** dynamically. For instance, if you want to put

tabulation size = 4 characters
Text mode font = roman type
(quasi-)TEX mode font = slanted type

you have only to insert a line of comment area with src2tex escape sequence of the form

```
{\src2tex{htab=4 textfont=rm texfont=sl}}
```

Generally, src2tex escape sequence is define by

{\src2tex{htab=*integer* textfont=*font type1* texfont=*font type2*}}

where *font type1* and *font type2* are equal to one of the following words respectively:

```
bf, it, rm, sc, sl, tt .
```

The meanings of the above key words would be self-evident (*cf.* [9]). For practical examples, it would be better to read SCHEME source program *farmer+hen.scm* and PS file *farmer+hen.scm.ps* which is actually generated by issuing the following commands:

```
% src2latex farmer+hen.scm
% latex farmer+hen.scm.tex
% dvi2ps farmer+hen.scm.dvi > farmer+hen.scm.ps
```

Another dirty trick is

`-<`*n*`>`

option of **src2tex** and **src2latex**. When you are in debugging mode, you usually need line numbers and you sometimes want to restrict page length. If you type

```
% src2tex -<n> source-file-name
% src2latex -<n> source-file-name
```

your source file is translated into debugging format. For example,

```
% src2tex -35 sqrt_mat.red
% tex sqrt_mat.red.tex
% dvi2ps sqrt_mat.red.dvi > sqrt_mat.red.ps~
% psnup -4 sqrt_mat.red.ps~  > sqrt_mat.red.ps
```

gives the *sqrt_mat.red.ps* file.

Acknowledgement

The authors would like to express their gratitude to
Andrew Allen ANDA
Mitsuru AOYAMA
Youzou DEGUCHI
Yoriaki FUJIMORI
Shinichi HATAKI†
Norihiro ITOH
Yoh ITOH
Hiroshi KAI
Azuki KANEKO
Shin MAEDA†
Takashi NAKAYAMA
Naoto NIKI
Kiyoko NISHIZAWA
Sakuro OZAWA
Issei SUZUKI
Hitoshi UCHIDA
Kenji YAMADA
Masayuki YAMASAKI‡
for giving them invaluable criticisms and suggestions.

References

† one of the authors of versions 2.8 and 2.9
‡ one of the authors of versions 2.10 and 2.11

[1] K. Amano, S. Hataki and S. Maeda, A text converter src2tex : Basic, C, C++, Fortran, Lisp, Pascal -> TEX (*to appear*).
[2] R. Baecker and A. Marcus, Human Factors and Typography for More Readable Programs, *Addison-Wesley, Reading, MA*, 1990.
[3] J. December, Presenting Java, *Sams.Net Publishing*, 1995.
[4] J. W. Eaton, Octave: A High-Level Interactive Language for Numerical Computations, Edition 1.1 for Octave version 1.1.1, 1995.
[5] B. Fuchssteiner and others, MuPAD Multi Processing Algera Data Tool User's Manual Draft Version, *Bennon Fuchssteiner, Automath, University of Paderborn, Paderborn, Germany*, 1992-95.
[6] A. C. Hearn, REDUCE User's Manual, version 3.5, *RAND Publication*, CP78(Rev. 7/94), 1994.
[7] B. W. Kernighan and D. M. Ritchie, The C Programming Language, *Prentice Hall*, 1978.
[8] D. E. Knuth, Literate Programming, *The Computer Journal*, **27** (1984), 97-111.
[9] D. E. Knuth, The TEXbook, *Addison-Wesley*, 1984.
[10] P. Mohseni, Maple in Action, An Introductory Handbook, *Iowa State University*, 1995.
[11] M. Noro and T. Shimoyama, Asir User's Manual, Edition 2.0 for Asir-940420, *Fujitsu Laboratories Ltd.*, 1994.
[12] J. K. Ousterhout, Tcl and Tk Toolkit, Addison Wesley Publishing Company, Inc., 1993.
[13] K. Sigmon, MATLAB Primer, Second Edition, *University of Florida*, 1992.
[14] G. L. Steele Jr., Common Lisp The Language, 2nd edition, *Digital Equipment Corporation*, 1990.
[15] D. B. Thompson, The Literate Programming Frequently Asked Questions List, *comp.programming. literate newsgroup*, 1995.
[16] S. Wolfram, Mathematica, A System Doing Mathematics by Computer, Second Edition, *Wolfram Research*, Inc., 1992.

```
* ____________________________________________________________________________ */


/* ______________________________ src2tex.c ______________________________ */


#include <stdio.h>
#include "src2tex.h"

extern void get_fnames();
extern void init_lang_flag();
extern void open_files();
extern void close_files();
extern void text2tex();


/* ____________________________ main function of src2tex ____________________ */


main(argc, argv)
int argc;
char **argv;
{
    char *cptr[2];                  /* character pointers of file names          */
    FILE *fptr[2];                  /* stream pointers of input/output files     */

    get_fnames(argc, argv, cptr);   /* get file names from the command line      */
    init_lang_flag(cptr);           /* initialize language flags                 */
    open_files(cptr, fptr);         /* open input and output files               */
```

```
    text2tex(cptr,fptr);                /* Text→TeX translation                  */
    close_files(cptr, fptr);            /* close input and output files          */
    exit(EXIT_SUCCESS);
}
```