Poor Man's Chinese in TEX

(traditional character version)

Thomas Ridgeway
Humanities and Arts Computing Center
University of Washington

This little document is intended as an illustration of and preliminary guide to *poor man's Chinese* [**pmC**]. **pmC** is a temporary expedient for printing texts with Chinese via TEX until such time as we have a well-thought-out system (and fonts) for handling Chinese.

pmC has a small number of virtues; among these are:

- **pmC** is available now.
- **pmC** is available free of charge.
- **pmC** is available now.
- **pmC** is device-independent (although it must be admitted that you will get relatively more pleasing results on less sophisticated printers)
- **pmC** is available now.
- **pmC** is sufficiently simple-minded you will not need to be a rocket scientist to make modifications for your own use.
- **pmC** is available now.
- **pmC** uses fonts mechanically faked-up from dot matrix quality fonts; if you want you can incorporate new characters of your own design using a bitmap font editor which *may* be easier to use than METAFONT.

For those who might be previously unacquainted with TEX, let us put forth a few of the advantages of TEX which are inherited by **pmC**: it can run on almost any modern computer; it can print to any of a large number of printers/typesetters; it can print (if desired and suitably instructed) elaborate formatting of the text; it is available at very low cost, or free (which is an instance of very low cost).

The disadvantages are at least equally compelling:

- **pmC** is crude and unlikely to ever be greatly improved as regards the quality of its Chinese.
- **pmC** uses emulation-in-METAFONT techniques to produce a dot-matrix font on your output device. No matter how talented your printer may be, the Chinese output by **pmC** will looked like it was stripped in from a dot-matrix printer.
- **pmC** is crude and unlikely to ever be greatly improved as regards the quality of its Chinese.
- **pmC** uses a large number of fonts which occupy a finite and measurable quantity of diskspace for each font. Your screen previewer had *better* be able to use

the same resolution your printer uses, or you may experience a little tightness around the disk. You probably will not want the fonts in multiple magsteps (they will just get uglier if you enlarge them).

- **pmC** is crude and unlikely to ever be greatly improved as regards the quality of its Chinese.
- **pmC** does not do slanted, bold, or other fancy Chinese although one could in principle. However, insofar as even a minimal installation requires 88 fonts *for Chinese alone*, we must at this point quote The Master:

  > the format of a *char_node* allows for up to 256 different fonts and up to 256 characters per font; ... TEX intended for Oriental languages will need even more than 256×256 possible characters when we consider different sizes and styles of type. Donald E. Knuth, *TEX: The Program.* Addison Wesley, Reading MA: 1986. (Volume B of *Computers and Typesetting*). §134, p. 57.

- **pmC** is based on TEX (hooray!), not on JTEX (hooray!) nor any not-known-to-me to-be-extant ChiTEX (huh?) which might know how to set type vertically, handle thousands of fonts, make doughnuts, and do all the other things that we might want software to do. **pmC**, therefore, does not set type vertically, handle thousands of fonts, emulate Spads or Spitfires, or any of the other real neat stuff.
- **pmC** may take a long time to set up if you need to make your own fonts. Preparing *one* set of fonts on a NeXT computer for use at 400 dpi required a METAFONT run of over 18 hours. The METAFONT source code files for the fonts occupied 17 megabytes of disk space when all were stored. If one looks at the simple-minded brute-force METAFONT code which the **pmC** fonts use, one may roughly estimate that a staggering number of computations (probably even more than eight) are needed to emulate a dot-matrix font through this method. There might, indeed, be a better way to do it. On the other hand, Pac-Man worldwide has consumed several hundred million times the number of CPU cycles consumed by compiling **pmC** fonts, so we are not exactly talking about wasting a scarce resource, are we? You will not ordinarily wish to make up the fonts more than once, assuming a relative absence of masochism in your personality makeup, nor will you probably want to keep the METAFONT source code permanently *since you can mechanically regenerate it.*

How to use **pmC**

Assume a text editor capable of enabling entering text in BIG-5 encoded Chinese characters. Assume a text editor capable of editing text in "normal" Latin characters. Use the first text editor to prepare a plain text file containing the Chinese text; *plain text file* means that there are no tricky codes in the text file which are

intended to be used only by the program that produced the file—"ASCII file" frequently conveys what we mean by "plain text file", but by virtue of having encoded Chinese, your file will not be ASCII. Use the second text editor to enter the Latin-letter strings providing instructions to TEX; (if your first text editor is o.k. for Latin too, you can continue using it for this step! — but watch out, the JIS and BIG-5 encoding schemes have some Latin characters (graphemes) included in the two-byte code space: we need "normal" Latin (one-byte coded) text). On a separate line before the first line of Chinese, put a line saying "`\input pmCb \beginBchinese`". Pmc.tex is intended to be the name of a file containing the basic set of instructions for **pmC**. "`\beginBchinese`" is a **pmC**-defined magic-word signifying that everything until an "`\endBchinese`" is to be interpreted as Chinese, if possible. Using whatever text editor you want, incorporate text in whatever other language you want. Other languages will conflict with Chinese in **pmC** only if they use character codes in the range 161 to 254; such languages should be separated from Chinese by enclosing the Chinese in "`\beginBchinese`" "`\endBchinese`" structures. Create a new last line saying "`\bye`". Put a blank line between paragraphs in the text, and use any other TEX commands with which you may be familiar to further adorn your text. Save the file, and say the appropriate words to run TEX on your file, preview it, print it, or whatever.

What can go wrong

Well, for one, you may not have an editor for entering Chinese. [Sorry; **pmC** has nothing to offer you if you have no way to compose your text.] Or, for another, your editor may not use BIG-5 encoding. [Easy: convert your text to BIG-5 encoding! (How? That's not my problem; or, to be a *little* more gracious about it, there are some converters out there, see if you can find one ....)] You may not know "any other TEX commands with which you may further adorn your text". (Careful study will reveal this phrase to be a direct quotation of nothing except itself.) [So take a class.] Something else goes wrong. [Fix it.] You don't have a copy of **pmC**. [Get one.]

How to get **pmC**

From a friend is good. If you have no friend who has **pmC** and is prepared to make a copy for you ... (that was supposed to be phrased as delicately as possible) ... you may get it through an *Established Channel*. There are a number of "TEX distributions" which may elect to incorporate **pmC** in their offerings of TEX and TEX-related software. You will very likely be able to get a copy from the NWCSC, distributors of TEX for Unix systems; they will have to charge you a fee, however, as even though they are non-profit they still have to pay their employees, pay phone bills, and all the rest of the drag with which I am sure you are familiar. You might also be able to get it electronically and free-of-charge via **ftp** from some place or other, but I couldn't tell you where. As I write, November 21, 1990, none of these alternatives which I have suggested will actually work, and there is no way that you can get **pmC** short of enrolling in a course of study at the University of Washington,

or you have a friend who has done so.

Are we really going to show any Chinese or what?

Oh yes, this is roughly how it looks:
今為止，大家所用的中文 編輯器，毋論用的是何輸入方式 都不過于 WYSIWYG (What You See Is What You Get) 類型 ，例如中文 WordStar 即聯想漢字 ， 天馬，飛馬，五筆字型。(Please note that we may intermix all kinds of other stuff in with the Chinese, such as Родина, Москва, Киев.) 這些軟件有它們好的一面， 即 直觀性強。它們大都用于 IBM 兼容機與 Macintosh 個人微機， 并需 特殊的中文操作系統來啓動機器。然而，它們的弱點卻是不可彌補的， 即輸入速度太慢而且太費腦力和眼 力。想想看，當鍵盤 輸入英文時

你只需機械地揮動手指，眼睛和頭腦 全力貫注在屏幕上出現的文章 十分輕松， 愉快，同時也能保証出好 文章。但是用中文 WordStar 時，你的 頭腦和眼睛卻要繁忙地碾轉在打字 、選字、和文章內容之間，如連續 兩小時如此作業，你將覺得腰酸 眼痛腦力竭盡，毫無樂趣可言。下回一提起打 中文你便有一種恐懼感。本人三年 以來使用中文軟件的結論便是如此。 那末 ChText 用的又是甚麼方法呢？ 首先 ChText 不是個編輯器，而是個 拼音 .

NOTE: This version of Thomas Ridgeway's document *Poor Man's Chinese in TEX* has been prepared by Stephen G. Simpson, for use with Simpson's new TEX input file, `pmCb.tex`. Simpson's file is a modification of Ridgeway's original `pmCs.tex`. Ridgeway's file enabled TEX to handle GB coded, simplified (mainland) Chinese characters. The new file does the same for BIG-5 coded, traditional (Taiwan) Chinese characters.

Simpson (`simpson@math.psu.edu`) would like to acknowledge the help and inspiration of Mark Leisher (`mleisher@nmsu.edu`) and Thomas Ridgeway (`ridgeway@blackbox.hacc.washington.edu`). The following Dirty Trick wizards contributed some ideas which resulted in the elimination of certain bugs: Raymond Chen (`raymond@math.berkeley.edu`), Victor Eijkhout (`eijkhout@csrd.uiuc.edu`) and Kresten Krab Thorup (`krab@iesd.auc.dk`). Any remaining bugs in `pmCb.tex` are Simpson's sole responsibility.

Last revised 27 October 1991