LATEX-ing

Linear Algebra

JIM HEFFERON

2003-MAY-05

This document contains material about using Jim Hefferon's undergraduate textbook *Linear Algebra*. This book is freely available on the web.

1 Summary of *Linear Algebra.*

Linear Algebra covers the material usually done in a three or four hour a week, one semester US course: solving linear systems, vector spaces, linear maps, determinants, and eigenvalues and eigenvectors. The pedagogical effort of the book is to help a student make the transition from calculus to upper division mathematics. Therefore the presentation stresses careful motivation, many examples, clarity in the proofs, and extensive exercise sets. The Preface contains a more detailed description.

1.1 Best When Viewed By . . .

The simplest way to have a look is to go to `http://joshua.smcvt.edu/linalg.html`. The files there can be viewed with Acrobat Reader. (You probably have Reader already installed on your computer but in any event you can get it for free from `http://www.adobe.com`. You can also view this material with *xpdf* or with the PostScript viewer *GhostView*.)

Alternatively, you can FTP to `joshua.smcvt.edu/pub/hefferon/book` and get `book.pdf`.

If you get both the book and the answers, name the first `book.pdf` and the second `jhanswer.pdf` and put them in the same directory so that the interfile hyperlinks work (clicking on a question gives you the answer, and clicking on the answer gets you the question).

1.2 The Materials are Free

One aspect of this book that is a bit different is that the materials are freely available, in source form.

That means, first of all, that you are free to download the book. You are free to use it yourself, and, if you are a mathematics instructor, you are also free to have students in your class use paper or electronic copies as a course text, either the main text or an auxilliary text.

Besides being free to use the output form of the text, you are also free to use the source form. Some examples of how this can be helpful are: an instructor can add a few favorite exercises here and there, or perhaps include a new Topic, or adjust the development in a subsection, or even rewrite a portion entirely. Another example is: someone who finds an error, say in the answer to an exercise, can send me the correction (i.e., a 'diff file').

With the source, and the instructions below on using the macros, these are all straightforward (meaning that with the computer work largely set up for you, and with this mass of examples to copy, jobs like these become mostly about knowing the mathematics rather than being dominated by the necessity of mastering computer details).

If you do make some adjustment, and you feel that you are able, your contribution of work back to the project would be welcome. I of course reserve the ability to not use some things but obviously all contributions that are used will be gratefully acknowledged.

2 Using the Source

To do more than read the material, you need to know that the files here are for the TEX system.

TEX is a typesetter which uses a markup language — it is not a word processor. Instead, you intersperse commands and text: for instance, the source '`Consider the vector space $V$`' will set 'Consider the vector space' as regular text, and then set the 'V' as math text (that's what surrounding text with `$`'s does). I used it for this project because I know of no other system that can do the job. In particular, the output is the best that I have ever seen from a personal computer. It especially shines at mathematics.

This book is written using the LaTeX macro system for TeX. The 'macro system' modifier means that commands have been added to make it easier to use the underlying TeX. For example, LaTeX simplifies the numbering of theorems, indexing, and cross-references.

2.1 Starting TeX-ing

Many people find that using the TeX system gives rise to a lot of head-scratching. As with any powerful tool, there is a lot to learn, and I can't in this document try to teach you TeX or LaTeX.

2.1.1 What you need to know

What I can do is to suggest that if you are a newbie then you could start with the *Short Guide to LaTeX2e* at `http://www.ctan.org/info/lshort2e`, which might be enough for you to get through making small changes in the materials here. After that, get *LaTeX: A Documentation Preparation System* by Lamport (ISBN 0-201-52983-1), and *The LaTeX Companion* by Goosens and others (ISBN 0-201-54199-8). Most university libraries will have both. You may also find useful the original book on TeX by its developer, *The TeXbook* by Knuth (ISBN 0-201-13448-9). There are many other good books and in particular people often recommend the one by Kopka and Daly (which I happen never to have used so I don't have an ISBN handy).

The graphics here are in MetaPost. For that language you need the manual from `http://cm.bell-labs.com/who/hobby/MetaPost.html` and also *The MetaFont Book* by Knuth (ISBN 0-201-13444-6).

You should also be aware of three resources. The first is the Comprehensive TeX Archive Network, CTAN, which is a group of Internet sites that cooperate to keep a very large collection of useful TeX material all in one place. There are sites all over the world, but you can get started at `http://www.ctan.org`. The second resource is that there are a number of Users Groups. Here also, all over the world there are groups, but a good single place to start is `http://www.tug.org`. If you find TeX and friends useful, join your local group. Finally, the third resource is the usenet group `comp.text.tex`. This is a mailing list that you can read, search, or post to, from `http://groups.google.com`. If you have a question, you've exhausted your background trying to make it out on your own, and you can't find the answer by searching old posts, try posting there yourself. It is a very friendly group, helpful and polite.

2.1.2 What you need to have

If you are just beginning then the first thing that you need is a distribution of TeX and friends (that is, you need a collection of all of the associated programs and other materials that together make this system so powerful). For information on where to get such a distribution, whether free or commercial, for various platforms, see `http://www.tug.org` under 'implementations'.

If you aim to modify or add material to the book and are unfamiliar with TeX then a good way to get started is to get this document to come out. These work for me under Linux: first 'latex 1st_readme', second 'dvips -P www -o 1st_readme.ps 1st_readme', and third 'gv 1st_readme.ps'. If that works — if the third command brings up a window with the output from this file — then you know your setup is at least partially correct. However, I must say that each TeX distribution has a slightly different approach and, as with TeX in general, I'm sorry, I can't tell you how each different distribution is set up exactly. You'll have to find that information in the distribution documentation.

What good is this 'How to Run It' file if it must be run? It builds character. Anyway, if you are having trouble even getting started, or only want a quick look, I've included the output material in the file `1st_readme.pdf`, which is readable with Acrobat Reader.

2.2 The Book's Files

The book's source material comes in a number of separate files.

file name	what it is about
book.zip	Material for the book as a whole and styles
front.zip	Front matter (e.g., the Preface) and back matter (the Appendix)
gr.zip	Chapter One, on Gaussian reduction
vs.zip	Chapter Two, on vector spaces
map.zip	Chapter Three, linear maps
det.zip	Chapter Four, on determinants
jc.zip	Chapter Five, on eigenvalues and eigenvectors

Each of these `.zip` files is a set of other files (it is an archive). Unzip it with `pkunzip`, or `winzip`, or the free material from the Info-zip folks.

After you unzip the above files, you will have in your directory many `.tex` files, a few `.mp` files, and a few other types such as LaTeX style files `.sty`. The `.tex` files are for LaTeX and the `.mp` files are for MetaPost for the graphics (a few graphics are also in Encapsulated PostScript `.eps` form).

Before you get the book going you need to have the graphics available. (Before typesetting a page, TeX needs to know how big the graphic is to determine how much room is left for text.) First, invoke MetaPost on the file `dotprod.mp`. On my system 'mpost dotprod.mp' does the job. It produces a file `dotprod.1`. This file is very simple; it contains only one graphic, the dot in dot product, and so if anything will work it should be this. That is, it is a good test.

Next invoke MetaPost on the five chapter files `mpost ch1.mp`, ..., `mpost ch5.mp`, and `mpost appen`. You'll get files whose names are `ch1.1`, `ch1.2`, etc. There should be a lot of them.

If that works then you are on a roll. Run LaTeX on the book. On my system the command 'latex book' works. The first time through should produce an alarming number of warning messages. Do it a second time. The number of messages should drop; what's happened is that the first time through LaTeX warned you about cross-references while it was saving them to a file, and the second time through it found them in the file so it did not warn you that they were not found. Make the index `makeindex -s book.isty -p odd book.idx`. LaTeX the book a couple more times. Then convert it to PostScript; I say 'dvips -P www -o book.ps book' (the '-P www' makes it use outlined fonts, which are better for reading on a screen). Then I can read the book using the *GhostView* program, with 'gv book.ps'.

To get the answers, do the same steps on *jhanswer.tex*. That is, 'latex jhanswer' done twice, followed by 'dvips -Pwww -ojhanswer.ps jhanswer', and followed by 'gv jhanswer' works for me.

2.2.1 When Good Systems Do Things That Are Bad

What can go wrong? Almost anything. As Tom Wolfe quotes the early astronauts, "It can blow at any seam." Discouraging, isn't it?

The MetaPost-ing can fail in a number of ways. First, you need the *3D* material from CTAN. Second, you need all of the MetaPost material that I've generated (e.g., *venn.mp* for Venn diagrams), although you'll have gotten that from me unless I've made a mistake. Third, typesetting the labels, etc., may result in MetaPost giving you a LaTeX error. The thing about this failure is that the error message will say that the first label is bad. But it isn't the first label; it is likely somewhere much further down the file. To figure it out you must run 'latex mpxerr.tex' (this oddly-named file is where MetaPost writes the labeling lines for them to be automatically LaTeX-ed). Then you'll get an error message that at least points to the correct line, and presumably a little staring at it will give you a clue. (I've wasted any number of hours looking at line one when the problem was somewhere else entirely.)

LaTeX-ing the book can then give you a wide variety of troubles.

It is a big book, and I use a number of styles. If you are having trouble that seems to come from, say, *color.sty*, your first try is to go to CTAN and download the latest. In particular, I find that *hyperref* has been a frequent changer over the course of time that I have been working on the text (understandably, obviously).

If you are having trouble with a part, see what commenting it out does. Go into *book.tex*, for instance

and comment out that part of the `includeonly` material (watch your commas! if you say ',vs3%,' instead of ',vs3,%' then you may find LaTeX complaining about not finding *vs3map1.tex* since *map1* starts the next line).

One area that can be annoying is that errors in the answer file do not tell you the line number in the original source file. Instead, they tell you the line number in the source file for the answers (probably *bookans.tex*). That file was written when you LaTeX-ed the book. So don't edit that file since your changes will disappear the next time you do the book again. Instead, you can look up that line in the answer's source, then look for that same line in the original source. Edit that one. Then LaTeX the book again. Thank goodness for fast computers. (Getting the line numbers from the original source into *bookans.tex* somehow would be a fine project for someone. Hint, hint.)

One trouble that I've had comes when switching between a *hyperref*-ed and non-*hyperref*-ed versions. LaTeX complains about not liking the cross-reference file (something about "fifthoffive"). My solution is to delete all the cross-reference files (under Linux, `rm -f *.aux *.toc`), and then rerun the LaTeX command line twice. Not very elegant, I know.

2.2.2 Options

I have set up a few options.

Set the DVI driver from the command line with '`latex "\def\dvidrv{dvipdfm}\input book"`' (the DVI driver matters to the *graphics* package). If you do not set this then the package *dvidrv.sty* that I provided tries to guess it from your *color.def* file, and failing that uses *dvips*. To set your own default you can change the line in *book.tex* to read, say, '`\usepackage[dvipsone]{dvidrv}`'.

Similarly, you can decide from the command line to have the output file hyperlinked (by using the package *hyperref*). Say '`latex "\def\hrefout{yes}\input book"`'. Anything other than 'yes', even 'Yes' or 'y', will leave the file not hyperlinked.

You can also use the pbsi font in a similar way: '`latex "\def\pbsiout{yes}\input book"`' will use the font on the cover page and on chapter pages (you probably need to download this font separately from a CTAN).

These options combine. I use '`latex "\def\dvidrv{dvipdfm}\def\hrefout{yes}\input book"`' to produce output for Reader. Some options options take some tweaking (*dvips* will work if you ask for hyperlinking, but you need to call it specially to get the best results: '`dvips -Ppdf -G0 -obook.ps book`', I think, can be distilled to give a sound `.pdf` file).

All those options apply to the answer file, and one more. When you LaTeX the book you can either choose to have all the answers output to a single file (the default filename is *bookans.tex*), or to have the recommended answers output to one file *recans.tex* and the other answers output to another *otherans.tex*. The default is one big answer file. To switch it, look for a line early in *book.tex* that says `\usepackage[single,write]{bookans}` and change the '`single`' to '`double`'. Then to read in the file of interest, invoke the answer file with, for instance, '`latex "\def\ansfile{recans}\input jhanswer"`'. The default here, matching the book's default, is to use the one big file *bookans.tex*.

2.2.3 Summary

These steps work for me. I make a new directory *book* and change into it. I `unzip linear_algebra`, and then also unzip each sub-archive.

These steps will give a perfectly usable version, suitable for printing.

`mpost dotprod.mp`	Make the dot product symbol.
`mpost ch1.mp ... mpost ch5.mp, mpost appen.mp`	Make the line art.
`latex book`	Produce the book's reference files.
`makeindex -s book.isty -p odd book.idx`	Alphabatize index.
`latex book`	Produce the book's .dvi file.
`dvips -Pwww book`	Produce the book in PostScript form.

To get the version on the web I substitute two commands for the final two on the above list. This

```
latex "\def\dvidrv{dvipdfm}\def\pbsifont{yes}\def\hrefout{yes}\input book"
```

produces the book's .dvi file (I had to get the *dvipdfm* DVI driver and the *BrushScriptItalic* font from CTAN; see the final section of this document). Then

```
dvipdfm book
```

gives the PDF output. I then run the same two commands with *jhanswer* instead of *book* to get the answers.
All these steps are in the unix shell script `make_book_for_web.sh`.

3   Guide to the Macros

You can use the same macros to write your materials as are used in the text. For example, when you make up an exam and you want to refer to the vector space of polynomials of degree three, use `$\polyspace_3$`, avoiding students' confusion and also simplifying your day. This section describes how to use those macros.

First, the traditional disclaimer. In developing the book, I got to know a little bit about how to write TEX and LATEX macros — but I didn't get to know a lot! If you can improve what is here, I'd welcome that contribution.

3.1   Linear algebra macros

The macros that seemed to me to be specific to typesetting the linear algebra material are in the file `linalgjh.sty`. For instance, here are macros for displaying systems of linear equations, matrices, row and column vectors, special vector spaces, linear maps, etc.

Note that the book uses the *amsmath* package, including the commutative diagram extension *amscd*, so all of those wonderful macros are available. In particular, displayed equations are shown with `equation*` (the * makes them not-numbered) and all of the alignment structures like `align` and matrix structures are available.

Another style that gets loaded is *mathrsfs*, for the Ralph Smith Formal Script fonts to make, for instance, the script R for the range space.

linsys Make a linear system, in such a way that the columns line up. Here is an example of a three-unknowns system; you want to do this only as a displayed equation.

```
\begin{equation*}
  \begin{linsys}{3}
     x &+ &3y &+ &a &= &7 \\
     x &- &3y &+ &a &= &7
  \end{linsys}
\end{equation*}
```

If a row is without some of the variables, be sure to nonetheless add the appropriate &'s.

```
\begin{equation*}
  \begin{linsys}{3}
     x &+ &3y &+ &a &= &7 \\
       &  &3y &  &  &= &7
  \end{linsys}
\end{equation*}
```

5

In the special case that a row starts with a negative sign, do not use &−. That is, do not start the second line below with `&- &3y`.

```
\begin{equation*}
  \begin{linsys}{3}
     x &+ &3y &+ &a &= &7 \\
       &  &-3y&+ &a &= &7
  \end{linsys}
\end{equation*}
```

Remarks.

1. In the exercises I might have three or four linear systems on a horizontal line and to get them to line up (to be `t`-aligned) I included an optional argument governing that vertical alignment.

2. *LATEX note.* The variables are put in the columns right-justified, while the additions or subtractions are centered. Between the columns is put 4/18-ths of an em (TEXbook, p. 167–170; that's a medmuskip).

3. *LATEX note.* The argument saying how many variables (which in the examples above is the 3) is there instead of some quite large number because it isn't as simple as $arg + 1$ times `rc`. But I wasn't sure, and might be convinced to change this by someone who knows what they are doing here.

grstep Denote a step of Gauss' method.

```
\begin{equation*}
  \begin{linsys}{2}
    x   &+   &y   &=   &0   \\
    x   &-   &y   &=   &1
  \end{linsys}
  \grstep{-\rho_1+\rho_2}
  \begin{linsys}{2}
    x   &+   &y   &=   &0   \\
        &    &-2y &=   &1
  \end{linsys}
\end{equation*}
```

Show more than one row operation at a time with `\grstep[2\rho_5]{\rho_1+\rho_3}` for two row operations, or `\grstep[2\rho_5 \\ 3\rho_6]{\rho_1+\rho_3}` for three. Swap two rows with `\rho_1\swap\rho_2`.

matrix structures Generic matrices are made with `pmatrix`.

```
\begin{equation*}
  \begin{pmatrix}
    1  &2  &3  \\
    4  &5  &6
  \end{pmatrix}
\end{equation*}
```

Note that there is no need to specify the number of columns; see the *amsmath* documentation.

But there are a few places where I needed something a little different. In particular, I needed augmented matrices:

```
\begin{equation*}
  \begin{amatrix}{2}
    1  &2  &3  \\
    4  &5  &6
  \end{amatrix}
\end{equation*}
```

produces a matrix that is two-by-three, with a vertical bar between the final and next-to-final columns. The argument 2 means that there are two columns before the vertical bar. I also sometime need matrices partitioned into columns:

```
\begin{equation*}
  \begin{pmat}{c|c|c}
    1  &2  &3  \\
    4  &5  &6
  \end{pmat}
\end{equation*}
```

produces a two-by-three matrix that has vertical bars separating the columns.

I make displayed determinant arrays with `detmat`

```
\begin{equation*}
  \begin{detmat}
    a  &c  \\
    b  &d
  \end{detmat}
  =ad-bc
\end{equation*}
```

and in-line determinants with `\deter{A}`.

vectors Make column vectors with `\colvec{1 \\ 2 \\ 3}`. Make row vectors with `\rowvec{1  &2  &3}`.

decimal point-aligned columns I use the *dcolumn* package:

```
\begin{equation*}
  \begin{aligncolondecimal}{3}
    15.12  &0.345
  \end{aligncolondecimal}
\end{equation*}
```

makes a column vector aligned on the decimal with room for at most three decimal places on the right.

strings A digit 3 in a square is `\digitinsq{3}`. Not surprisingly, a digit 3 in a circle is `\digitincirc{3}`. A sequence of strings is shown this way.

```
\begin{equation*}
  \begin{strings}{ccccc}
        \vec{e}_1 &\mapsto &\vec{e}_2 &\mapsto &\zero \\
        \vec{e}_3 &\mapsto &\zero
  \end{strings}
\end{equation*}
```

There are five `c`'s because the longest line is of length five.

names for things Note that there is a page (inside the book's cover) covering the notation conventions.

I tried to remember to make up macros to name things, rather than retype the thing each time I ran across it. Here is a list.

1. The reals `\Re`, the rationals `\Q`, the complex numbers `\C`, the integers `\Z`, and the natural numbers `\N` come out in the traditional "blackboard bold".

2. A vector is `\vec{v}_j`. The zero vector is `\zero`. The length of a vector is `\norm{\vec{v}}`, and the absolute value of a number is `\absval{r}`. An angle can be expressed in degrees as `$53\degs$`. The distance between two vectors is `\dist (\vec{v}_1,\vec{v}_2)`.

3. The dot product of two vectors `\vec{v}\dotprod\vec{w}`. Please note that (as it is set up coming from me) this is different than a `\cdot`.

4. The vector space of degree n polynomials `\polyspace_n`, and the vector space of n-by-m matrices `\matspace_{\nbym{n}{m}}`. The vector space of linear maps from V to W `\linmaps{V}{W}`.

5. The span of a set S of vectors `\spanof{S}`.

6. The row space of a matrix is `\rowspace{M}` and the columnspace is `\colspace{M}`.

7. A set is `\set{\colvec{x \\ y}\suchthat 2x+y=0}`. The union and intersection of the sets S and T is `S\union T` and `S\intersection T`. The complement of a set is `\complement{S}`.

8. The empty set is `\emptyset`.

9. A sequence, such as a string, is `\sequence{s_0,s_1,\dots,s_n}`. The concatenation of two sequences is `\cat{B_1}{B_2}`.

10. A basis is `\basis{\vec{\beta}_1,\dots,\vec{\beta}_n}`. The standard basis for real n-space is `\stdbasis_n`.

11. Isomorphic spaces is `V\isomorphicto W`.

12. The matrix representing a linear map h with respect to the bases B and D is `\rep{h}{B,D}`.

13. The size of a general matrix is `\nbym{n}{m}` while the special case of a square matrix is `\nbyn{n}`.

14. A map h's range space `\rangespace{h}` and null space `\nullspace{h}`, and generalized range space `\genrangespace{h}` and generalized null space `\gennullspace{h}`

15. The direct sum of two subspaces `V\directsum W`.

16. The function, as specified by its domain and codomain, is described by `\map{f}{D}{C}`. Its action on a single element x is `x\mapsunder{f} f(x)`. Two maps are composed with `\composed{g}{f}`. The identity map is `\identity`.

17. The projection of a vector into a subspace `\proj{\vec{v}}{S}`.

18. The restriction of a map to some subdomain is `\restrictionmap{f}{S}`.

19. The rank of a matrix `\rank (A)` and the nullity of a matrix `\nullity (A)`. The transpose of a matrix `\trans{A}`. The trace of a matrix `\trace (A)`. The adjoint of a matrix is `\adj (A)`.

20. The size of a box is `\size (B)`.

21. The signum of a permutation ϕ is `\sgn (\phi)`.

22. For the Topic on voting, voter preferences are indicated by `\votepreflist{7}{1}{5}` for the column vector, and `\voteprefloop{1}{2}{3}` for the circle (for this one, the first argument appears at the ten o'clock position, the second argument at six o'clock, and the third argument at two o'clock).

23. A generic field is `\F`.

aligned vdots To make `\vdots` come out inside a bunch of aligned equations, I use `\alignedvdots`.

```
\begin{align*}
  a_{1,1}x_1+\cdots+a_{1,n}x_n &= d_1          \\
                                &\alignedvdots \\
  a_{m,1}x_1+\cdots+a_{m,n}x_n &= d_m
\end{align*}
```

Otherwise the three vertical dots are not centered on the equals sign.

3.2 Book layout macros

The macros that do chapter and section headings, or cross references, or how the exercises are laid out, are in the file `bookjh.sty`.

theorem-like structures I have already declared the `theorem`, `proposition`, `axiom`, `definition`, and `notation` environments. These will come out shaded. To change this, or the color of the shading or its border, see also *shadethm.sty*

```
\begin{definition}
  A \definend{big} vector space is one where you can't  see the end when
  you are standing at the zero vector, even with your glasses on.
\end{definition}
```

I have also declared (not shaded): `lemma`, `corollary`, `example`, `remark`, `note`, and `comment`.

By the way, note the `\definend`; I use this for the term being defined (I though I would automate indexing, but I've not got around to that).

proofs Use this.

```
\begin{proof}
   Because I said so; that's why.
\end{proof}
```

The end-of-proof symbol can be changed as `\renewcommand{\qedsymbol}{$\box$}`. I avoid ever having a proof end with a displayed equation so the question of where to put the end-of-proof symbol in that case never arises.

footnotes Footnotes come out on the same page as the reference. I only use footnotes to suggest that a reader can look for more in the Appendix `\appendrefs{codomains}`.

graphics I am using MetaPost for all of the graphics and all of the DVI drivers that I am interested in handle *.mp*'s straight off, so I have not used the `inlinegraphic` and `scannedpicture` environments. Instead I say this.

```
\begin{center}
   \includegraphics{gr2.7}
\end{center}
```

Putting two graphics side-by-side, or putting parallel text, are all a bit messy. Either look at the document on CTAN in the `info` directory, or just mimic what I did somewhere.

exercises and answers The exercise portions of a subsection looks like you might expect.

```
\begin{exercises}
  \item
    First Exercise
    \begin{answer}
      Answer to the first exercise.
    \end{answer}
  \item
    Second Exercise
    \begin{answer}
      Answer to the second.
    \end{answer}
  \recommended \item
    Third Exercise, recommended.
    \begin{answer}
        Answer to the third.
    \end{answer}
 \end{exercises}
```

Exercises are, of course, numbered automatically, in the same sequence as the other numbered parts of the text.

My goal is to have all exercises have answers, even the proof exercises. Of course, the answers are not written directly to the text; they are written to one or two separate files. Use *jhanswer.tex* to print them.

Remarks.

1. Some of the exercises have parts. I use the environment `exparts`. The items in such an environment are `\partsitem`'s.

```
\recommended\item Do this one right away.
  \begin{exparts}
    \partsitem Do this first.
    \partsitem Do this second.
  \end{exparts}
  \begin{answer}
    \begin{exparts}
      \partsitem Answer to the first.
```

```
        \partsitem Answer to the second.
      \end{exparts}
    \end{answer}
```

I sometimes use `exparts*`, which leaves the items in a horizontal list, but it doesn't work very well (it sometimes causes TeX's `Underfull hbox` errors; this is not as easy to fix as using a paralist because I don't want the beginning of a part on one line and the end on another).

2. Any desired changes to the spacing, etc., of exercises is in `exerciselist`. See also the style for the answers in *bookans.sty*.

3. You can redefine `\recommendationmark` to change the mark used to denote recommended exercises. You can also mark puzzles with `\puzzle\item`. This can be combined with `\recommended`.

4. If you are putting in an exercise without an answer, your best bet is to put an answer environment that is blank. But really, you shouldn't do that. I have found that doing the answers to all of the exercises improved both the questions and the presentation greatly, since I often found that I needed a detail here, or some tweak there. (If what you want is that students can't read the answer in their answer list, that is a different matter. See the next two remarks.)

5. These answers are typically quite detailed, giving the work and not just the answer; I approached the answers expecting that the reader wants to learn and did not take the approach that the reader is trying to cheat. (*Remark.* As a practicing teacher, I am very aware that it is not a simple issue. Nonetheless, in my opinion, a reader is entitled to enough exercise and answer pairs that, having tried a good selection and not just read the questions and peeked at the answers, they can move on well-prepared to the next subsection. I figure that a dozen such pairs in each subsection is about right. Then, to provide a selection and also to allow someone a second crack at a problem type that gave them trouble the first time, I aimed for two dozen questions in each subsection. If this question/answer situation bothers you, I would very much welcome additional submissions of exercise sets for which students do not have access to the answers.)

6. As I've set it up, you can have the answers written to either a single file, or separate files for the answers to recommended exercises and the answers to the others. See the description above under 'Options'.

7. *LaTeX note.* You can use `\ref`'s inside answers, even though they are written to other files. The style for the answers reads in the reference files.

8. An answer that is taken from a cited source may not blend stylistically with the others (insofar as I have a style). I use the disclaimer `\answerasgiven`.

tfae A small environment for The Following are Equivalent-type lists.

computercode Another small environment, this time for displaying computer code, obviously. I intend to change this to use the very nice *listings* style when I get a chance.

indented An environment that is much like LaTeX's `trivlist` but which I thought fit better with the typographic look of the rest of the book.

clearemptydoublepage Taken straight from the *Companion*. It does what it says.

cross references I use `\nearbydefinition{def:BigThm}`-like references. This has the disadvantage that the referrer has to know what type of textual element it is referring to (in this case a definition), but LaTeX doesn't give you a way to know in this case (that I can see). Note that I try to make the `\label{def:BigThm}` moderately descriptive.

I have set up nearby cross references (meaning there is no need to reference the chapter number and the section number) for: `\nearbydefinition`, `\nearbyfigure`, `\nearbylemma`, `\nearbyexample`, `\nearbycounterexample`, `\nearbytheorem`, `\nearbycorollary`, `\nearbyexercise`, `\nearbyremark`, `\nearbynotice`, and `\nearbynote`. (I don't use most of these, but they are there anyway.)

topics Begin a new topic with `\topic{My Favorite Topic}`. The exercises come out looking a little differently automatically, because I couldn't think of a common treatment that was both convenient and consistent.

optional subsections A subsection is marked as optional with `\subsectionoptional` instead of LATEX's standard `\subsection`. This asterisk's the name in the Table of Contents. I have also followed the practice of beginning each optional subsection with a italicised comment that it is optional, but this is not part of the macro.

page headings This work is all done through the wonderful *fancyhdr* style. See the documentation for that package.

hyperlink style I have set the links to come out in blue. Change that by altering the

```
\usepackage{
          ..,
          linkcolor=blue,
          ..}{hyperref}
```

entry.

4   Making Your Exams

I have included an exam style that does what you'd think it would do, number the exercises automatically, automatically put headers and page numbers, and optionally allow exercises to have attached answers that are printed to a separate file.

Here is a sample exam, without answers. See the web page for more extensive examples.

```
\begin{exam}
  \item \pts{15}
    How much wood could a woodchuck chuck?
    \begin{enumerate}
      \item If he would chuck wood?
      \item If not?
    \end{enumerate}
  \item \pts{85}
    Justify the Axiom of Choice, since we all know it is true anyway.
\end{exam}
```

5   Bugs

These are known to me; I'd appreciate any help, of course.

2001-Oct-08 I can't figure out how to get *hyperref.sty* to have the indexing links both work and show the correct page number (they are advanced by ten, which I presume is length of the front matter). I've disabled the indexing links with `hyperindex=false`.

6 Extra: Installing the Brushscript Fonts

These are the steps that I followed to install the brushscript fonts on a fresh Debian Linux file system.

I first downloaded the source files from `http://www.ctan.org`. To do that, I clicked on `search`, typed *brushscript* in the box, and elected to download the entire directory as a .zip file.

Then, with the source files on my machine, as the superuser, I first made `/usr/local/share/texmf` for my non-teTEX files. Having a local tree ensures that I won't overwrite the material if I install a new standard tree.

Then I followed these steps.

1. `mkdir /usr/local/share/texmf/fonts` just as in `/usr/share/texmf`

2. `mkdir /usr/local/share/texmf/fonts/source`

3. `mkdir /usr/local/share/texmf/fonts/source/public`

4. `cp brushscr.zip /usr/local/share/texmf/fonts/source/public` to put it in place

5. `cd /usr/local/share/texmf/fonts/source/public`

6. `unzip brushscr`

7. `cd brushscr`

8. `make` as the brushscripts directions in the directory say

9. `mkdir /usr/local/share/texmf/fonts/tfm` for TEX font metrics

10. `mkdir /usr/local/share/texmf/fonts/tfm/public`

11. `mkdir /usr/local/share/texmf/fonts/tfm/public/brushscr`

12. `cp *.tfm /usr/local/share/texmf/fonts/tfm/public/brushscr`

13. `mkdir /usr/local/share/texmf/fonts/vf` for virtual fonts

14. `mkdir /usr/local/share/texmf/fonts/vf/public`

15. `mkdir /usr/local/share/texmf/fonts/vf/public/brushscr`

16. `cp *.vf /usr/local/share/texmf/fonts/vf/public/brushscr`

17. `mkdir /usr/local/share/texmf/tex/latex/brushscr` LATEX font definition files

18. `cp *.fd /usr/local/share/texmf/tex/latex/brushscr`

19. `cp *.sty /usr/local/share/texmf/tex/latex/brushscr` and LATEX styles

20. `mkdir /usr/local/share/texmf/fonts/type1` Postscript outline fonts

21. `mkdir /usr/local/share/texmf/fonts/type1/brushscr`

22. `cp *.pfa /usr/local/share/texmf/fonts/type1/brushscr` ASCII-encoded Postscript fonts

23. `mkdir /usr/local/share/texmf/dvips` tell the dvi drivers

24. `mkdir /usr/local/share/texmf/dvips/config`

25. `cp /usr/share/texmf/dvips/config/config.ps /usr/local/share/texmf/dvips/config/config.ps`

26. `cp pbsi.map /usr/local/share/texmf/dvips/config`

27. `emacs /usr/local/share/texmf/dvips/config/config.ps` edit the config file

28. I then found the lines that say, "This shows how to add your own map file. Remove the comment and adjust the name:" and the next line says `p +myfonts.map` (but it is commented out with a percent sign). I added a line just below that to say (without the percent sign) `p +pbsi.map`.

Finally, I ran the `texhash` command to tell TEX's path searching mechanism about the new files.