Contents

Preparation of PCMI Lecture Notes

John C. Polking

IAS/Park City Mathematics Series
Volume XX, XXXX

Preparation of PCMI Lecture Notes

John C. Polking

Introduction

These notes are provided in order to simplify and uniformize the preparation of the lectures notes for the Graduate Summer School at the Park City Mathematical Institute.

Since the notes for the various lecture series eventually appear in one volume, we want to ensure some uniformity of appearance. At the same time we want to make the preparation of the notes as easy as possible for the lecturers. To this end the American Mathematical Society (AMS) provides us with a document class file, `pcms-l.cls`. This is based on the AMS book document class, `amsbook.cls`.

This document is (rather artificially) prepared using `pcms-l.cls` in order to provide a sample of its use. The niceties of this document class are explained in the first lecture.

The second lecture is devoted to the use of $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX to typeset mathematics. Specific examples are presented so that this document can serve as a robust model of a PCMI lecture series.

In the third lecture we discuss the use of graphics in the lecture notes. In particular we explain the limits that the AMS puts on the use of graphics. We also offer some advice on how to meet those limits.

The fourth lecture contains a list of resources for the use of TeX and $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX.

Department of Mathematics, MS 136, Rice University, PO Box 1892, Houston, TX 77251
E-mail address: polking@rice.edu
I would like to thank Barbara Beeton of the American Mathematical Society for writing pcms-l.cls, the TeX document class in which these notes were typed.

LECTURE 1

The PCMI Document Class

The class `pcms-l.cls` is a LaTeX 2ε document class developed by the AMS. It modifies `amsbook.cls` to allow the use of the lecture format seen in this document. The lecture format is the preferred format for the PCMI lectures. However, if he or she prefers, a lecturer may present the material of the lecture series in an article format. Both formats are described here.

In order to use `pcms-l.cls` you will need a TeX installation on your computer and the class file `amsbook.cls` installed. The AMS TeX Resources Home Page (http://www.ams.org/tex/) is a starting point for both. It provides links to sources for TeX and LaTeX as well to sources of information about setting up a TeX installation. It also provides a link to the AMS-LaTeX collection. If your LaTeX installation is recent (after August 2004), you should already have `amsbook.cls` installed. If not, you should download and install the entire AMS-LaTeX collection.

1. The organization of a file in the lecture format.

Your lecture TeX file should follow the standard outline of a LaTeX 2ε file. We will describe that organization in what follows. You can find more detail in [**GR**] or [**HK**].

1.1. The document class declaration

The document should start with the document class declaration

```
\documentclass[lectures]{pcms-l}
```

The option `lectures` indicates that this file will use the lecture format. The alternative would be `article`.

You can also add other LaTeX 2ε options. In particular you might use the `draft` option to clearly mark the overfull lines which you will want to eliminate before submitting your manuscript. To do this use the declaration

```
\documentclass[lectures, draft]{pcms-l}
```

1.2. The preamble

This is the place to input the extra packages that you use, to set up your declarations for theorems, lemmas, definitions, etc., and to define your own macros.

1.2.1. *Packages.* Since `pcms-l.cls` is based on `amsbook.cls`, which automatically loads the `amsmath` and `amsfonts` packages, it is not necessary to input these. However, you will undoubtedly use others. In particular, if you have any graphics, you will want to use the `graphicx` package. Thus you will have a section that resembles

```
\usepackage{graphicx}
\usepackage{amssymb}
\usepackage{mathrsfs}
\usepackage{latexsym}
```

```
\usepackage{verbatim}
```

Please list only those packages you actually use.

1.2.2. *Theorem-like declarations.* One of the nicer features of LaTeX 2_ε is the ease with which these declarations can be made. Thus you might include a list something like

```
\theoremstyle{plain}
\newtheorem{thm}{Theorem}
\newtheorem{lem}{Lemma}
\newtheorem{prop}{Proposition}
\newtheorem{cor}{Corollary}

\theoremstyle{definition}
\newtheorem{def}{Definition}
\newtheorem{example}{Example}
\newtheorem{exer}{Exercise}

\theoremstyle{remark}
\newtheorem{rem}{Remark}
```

There are three different "theorem" styles available:

- `plain` – the text is in the italic font with extra space above and below.
- `definition` – the text is in the roman font with extra space above and below.
- `remark` – the text is in the roman font with no extra space above and below.

You can find more information about theorem-like declarations in [**AM**], [**AT**], [**GR**], and [**HK**].

1.2.3. *Numbering.* The above theorem-like declarations will result in each of the items being numbered separately in one string throughout the manuscript. More than likely this is not what you want. Many people think it is more natural to restart the numbering in each lecture and precede the number with the lecture number. This can be achieved by adding a counter option to each declaration. For example,

```
\newtheorem{lem}{Lemma}[chapter]
```

will do the job for your lemmas. (This is currently a bug in `pcms-l.cls`. The option should really be `[lecture]`, but `[chapter]` is what works.)

To number equations by lecture use the command

```
\numberwithin{equation}{chapter}
```

(Again we have the bug.)

Some people like to number theorems, lemmas, propositions, corollaries, and perhaps other things together in one series. This can be accomplished with the commands

```
\newtheorem{theorem}{Theorem}[chapter]
\newtheorem{lemma}[theorem]{Lemma}
\newtheorem{proposition}[theorem]{Proposition}
\newtheorem{corollary}[theorem]{Corollary}
```

It is essential that the Theorem declaration appear first in this list of commands.

1.2.4. *Macros.* Everyone has their own macros. These should all be included in the preamble. The primary reason for this is that it makes things a whole lot easier for editors. Have pity on us.

For example, in this document we use the macros

```
\newcommand\pcms{\texttt{pcms-l.cls}}
\newcommand\amsbook{\texttt{amsbook.cls}}
```

1.3. Preparation for the main part of the document

After the preamble we are almost ready for the mathematics. Some preparation is required, however. As an example, the main part of this document starts with the commands

```
\begin{document}

\frontmatter
% \tableofcontents

\mainmatter

\LectureSeries[PCMI Lecture Notes]%
{Preparation of PCMI Lecture Notes \author{John C. Polking}}

\address{Department of Mathematics, MS 136, Rice University, PO Box
  1892, Houston, TX 77251}

\email{polking@rice.edu}

\section*{Introduction}
```

1.3.1. *Document environment.* Everything from this point on is done in the document environment. That means that everything is placed between the two commands

```
\begin{document}
...
\end{document}
```

The command `\end{document}` should be the last line in the file.

1.3.2. *The front matter.* This simply amounts to

```
\frontmatter
\tableofcontents
```

The command `\tableofcontents` will cause a table of contents to be developed. That is not essential until the final document is assembled, but you might like to see it anyway.

1.3.3. *Main matter.* This starts with two commands

```
\mainmatter
\LectureSeries[short-name]{long-name \author{author-name}}
```

In `\LectureSeries`, `short-name` is a shortened name for the lecture series suitable for appearance in page headers, while `long-name` is the real name for the series. Notice how `\author{author-name}` appears within the curly brackets for `long-name`.

Next we need some information about the author. The commands

```
\address{author-address}
\email{author-email-address}
%%\subjclass[2000]{Primary}
%% \keywords{}
%% \date{}
%% \thanks{}
```

should be self-explanatory. The items that are commented out are optional. However, please provide your official mailing and email adresses.

1.4. Introductory material

Almost every lecture series begins an introductory section with a title like "Introduction", or "Preface", or perhaps something more inventive. Start such a section with the command

```
\section*{intro-name}
```

The asterisk will ensure that the section is not numbered. There can be two or more such sections.

1.5. Subdivisions of the document

Each lecture is originated with the command

```
\lecture{lecture-name}
```

In the compiled document each new lecture starts on a new odd-numbered page.

Within a lecture the author has available sections, subsections, and subsubsections. These are called with the commands

```
\section{section-name}
\subsection{subsection-name}
\subsubsection{subsubsection-name}
```

The effect of these commands is illustrated in this document.

1.6. The bibliography

This is prepared in the way that is standard in LaTeX 2_ε, and is explained in [**HK**].

2. The organization of a file in article format

The article format is essentially one lecture without a beginning `\lecture` command. There are only three differences about preparing a manuscript in the article format. We will only describe these few differences.

First of all, the document class declaration should be

```
\documentclass[article]{pcms-l}
```

The second difference is that the only subdivisions that are available are the section, subsection, and subsubsection. There is no lecture division. The major subdivisions of the manuscript are sections. The sections should be started with the command

```
\section[short section name]{section name}
```

The short section name is needed only if the section name is too long to fit in the header of the even numbered pages.

Finally, since there is no lecture division, we cannot number equations, theorems, etc., by lecture. A suitable modification is to do so by section. This requires that the theorem-like declarations should read

```
\newtheorem{theorem}{Theorem}[section]
```

Notice that we are using the counter `section` instead of `chapter` as we did in 1.2.3. It will also be necessary to add the command

```
\numberwithin{equation}{section}
```

so that the equations are numbered by section.

LECTURE 2

$\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX Examples

Here we will summarize the most important methods for putting mathematical formulas into a TeX document. At the risk of boring the experienced reader we will start with the easiest examples.

1. Math within text

The original way of doing this in TeX is to use the dollar sign symbol.

```
The commutative law says that $x+y = y+x$.
```

LaTeX provides two other ways,

```
The commutative law says that \(x+y = y+x\).
```

and

```
The commutative law says that \begin{math}x+y = y+x\end{math}.
```

Any of these three will produce

The commutative law says that $x + y = y + x$.

Most people think that the use of `$x+y = y+x$` is still the most convenient.

2. Single line displayed formulas

2.1. Unnumbered equations

The original TeX way is to use the double dollar sign

```
$$x+y=y+x$$
```

LaTeX again provides two other ways

```
\begin{displaymath}
  x+y=y+x
\end{displaymath}
```

and

```
\[x+y=y+x\]
```

In addition, $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX allows the use of the starred `equation` environment

```
\begin{equation*}
  x+y=y+x
\end{equation*}
```

Any of these four ways will produce the display

$$x + y = y + x.$$

2.2. Numbered equations

To add a reference number to an equation, use the unstarred `equation` environment

```
\begin{equation}
  \label{commlaw}
  x+y=y+x
\end{equation}
```

This will produce

$$x + y = y + x \tag{2.1}$$

The command `\label{commlaw}` allows us to easily refer back to the commutative law (2.1) with the command `\eqref{commlaw}`. Notice that we used `\eqref` and not `\ref`. The later command would provide the equation number without the parentheses.

3. Multiline displays

For this purpose LaTeX provides the `eqnarray` environment. This environment produces very bad spacing, and it is not recommended. To replace it, the AMS-LaTeX package adds the following six environments:

`align` `gather` `falign` `multiline` `alignat` `split`

With the exception of the `split` environment, each has both a starred and an unstarred form.

Clearly, there are a lot of options. So many that it is difficult to learn them all. However, I suggest that almost everything can be done quite easily using only the `align` and `split` environments in their starred and unstarred forms. This is demonstrated in the examples that follow.

3.1. Aligned equations with all equations numbered

This is produced by the unstarred `align` environment. For example

```
\begin{align}
  x + y &= y + x,\label{addcomm}\\
  xy &= yx.\label{multcomm}
\end{align}
```

produces

$$\begin{aligned} x + y &= y + x, && (2.2)\\ xy &= yx. && (2.3)\end{aligned}$$

Notice that `\\` is the end of line command, and that the ampersand `&` is the alignment character, which tells TeX where to match up the two lines. In order to get correct spacing, the alignment character should always be placed directly before the character you want to align on. Thus, you should use `&=` and not `& =`.

Again, we can refer to the commutative law for multiplication (2.3) with the command `\eqref{multcomm}`.

3.2. Aligned equations with no equations numbered

This is produced by the starred `align*` environment. For example

```
\begin{align*}
  x + y &= y + x,\\
  xy &= yx.
\end{align*}
```

produces

$$x + y = y + x,$$
$$xy = yx.$$

Thus the starred version `align*` omits the numbers, while the unstarred version `align` assigns a number to every line.

3.3. Displays with only some lines numbered

The way to achieve this is to use the `\notag` command. For example,

```
\begin{align}
    x+y&=y+x,\label{addcomm3}\\
    xy&=yx.\notag
\end{align}
```

produces

$$x + y = y + x, \tag{2.4}$$
$$xy = yx.$$

3.4. Equations too long for one line

The `split` environment was invented to handle these situations. It provides a convenient way to split such an equation into two or more lines with a single equation number nicely positioned. For example,

```
\begin{align}
  \begin{split}
    u(x,y) &= \sum_{n=1}^\infty a_n\sinh\left(\frac{n\pi y}a\right)
    \sin \left(\frac{n\pi x}a \right)\\
    &\quad +\sum_{n=1}^\infty b_n\sinh\left(\frac{n\pi(y-b)}a\right)
    \sin \left(\frac{n\pi x}a \right)
  \end{split}\label{solsimplap}
\end{align}
```

produces

$$\begin{split} u(x,y) &= \sum_{n=1}^\infty a_n\sinh\left(\frac{n\pi y}a\right) \sin \left(\frac{n\pi x}a \right)\\ &\quad +\sum_{n=1}^\infty b_n\sinh\left(\frac{n\pi(y-b)}a\right) \sin \left(\frac{n\pi x}a \right) \end{split} \tag{2.5}$$

Notice that the equation number is centered so that it clearly refers to both lines used to define u.

3.5. Aligned equations with one number for all

While `split` was invented to deal with splitting very long equations, it can also be used effectively to group equations. For example,

```
\begin{align}
  \begin{split}
    x+y&=y+x,\\
    xy&=yx.
  \end{split}  \label{commlaws}
\end{align}
```

produces

$$\begin{split} x+y&=y+x,\\ xy&=yx. \end{split} \tag{2.6}$$

Notice that there is one equation number, nicely centered between the two lines. We can refer to the commutative laws, (2.6), with the command `\eqref{commlaws}`.

3.6. More complicated alignments

Suppose we want to list the commutative laws followed by the associative laws and then the distributive law, with one number for each. This can be done with

```
\begin{align}
  \begin{split}
    x+y&=y+x,\\
    xy&=yx,
  \end{split}\label{commlaws2}\\
  \begin{split}
    (x+y)+z & = x+(y+z),\\
    (xy)z &= x(yz),
  \end{split}\label{assoclaws}\\
  x(y+z) &= xy + xz.\label{distlaw}
\end{align}
```

We get

$$\begin{split} x+y&=y+x,\\ xy&=yx, \end{split} \tag{2.7}$$

$$\begin{split} (x+y)+z&=x+(y+z),\\ (xy)z&=x(yz), \end{split} \tag{2.8}$$

$$x(y+z)=xy+xz. \tag{2.9}$$

Notice that TEX maintains the alignment even outside of each `split` environment.

3.7. Formulas in two or more columns

The `align` environment allows the use of more than one alignment character (`&`). This enables us to display formulas in more than one column. For example,

```
\begin{align}
  x + y   &= y + x,   &    xy &= yx,\label{commlaws3}\\
  (x+y)+z &= x+(y+z), & (xy)z &= x(yz).\label{assoclaws2}
\end{align}
```

produces

$$x+y=y+x, \qquad xy=yx, \tag{2.10}$$

$$(x+y)+z=x+(y+z), \qquad (xy)z=x(yz). \tag{2.11}$$

Notice the different functionality of the odd-numbered and even-numbered ampersands. The odd-numbered ampersands are alignment characters, while the even-numbered ones are column separators.

4. Theorem, lemmas, corollaries, definitions, etc.

Having defined our theorem-like macros in the preamble, it is now easy to use them. For example,

```
\begin{definition}\label{compact}
  A set $K$ is compact if every open cover of $K$ has a
  finite subcover.
\end{definition}
```

produces

Definition 2.1. A set K is compact if every open cover of K has a finite subcover.

If you want to add a name to a definition you can do so in square brackets. Thus

```
\begin{definition}[Compact set]\label{compact2}
  A set $K$ is compact if every open cover of $K$ has a
  finite subcover.
\end{definition}
```

produces

Definition 2.2 (Compact set)**.** A set K is compact if every open cover of K has a finite subcover.

The label allows us to refer to Definition 2.1 as `Definition~\ref{compact}`. (The tilde (~) is used to prevent a line break between Definition and 2.1. It is not needed here, but it is good to be sure.)

Theorems, etc., are handled similary. Here are some examples.

```
\begin{theorem}[Existence of solutions]\label{exist}
  Suppose the function $f(t,x)$ is defined and continuous on the
  rectangle $R$ in the $tx$-plane. Then given any point
  $(t_0,x_0)\in R$, the initial value problem
  $$x' = f(t,x)\quad\text{and}\quad x(t_0) = x_0$$
  has a solution
  $x(t)$ defined in an interval containing $t_0$.  Furthermore, the
  solution will be defined at least until the solution curve
  $t\to(t,x(t))$ leaves the rectangle $R$.
\end{theorem}
```

produces

Theorem 2.1 (Existence of solutions)**.** *Suppose the function $f(t,x)$ is defined and continuous on the rectangle R in the tx-plane. Then given any point $(t_0,x_0)\in R$, the initial value problem*

$$x' = f(t,x)\quad\text{and}\quad x(t_0) = x_0$$

has a solution $x(t)$ defined in an interval containing t_0. Furthermore, the solution will be defined at least until the solution curve $t\to(t,x(t))$ leaves the rectangle R.

To add a remark, use the `remark` environment.

```
\begin{remark}
  Notice that Theorem~\ref{exist} says nothing about the
  uniqueness of the solution.
\end{remark}
```

produces

Remark 2.1. Notice that Theorem 2.1 says nothing about the uniqueness of the solution.

For the proof, use the `proof` environment.

```
\begin{proof}
  This environment will put the proof in roman type, and
  will display an end of proof symbol.
\end{proof}
```

produces

PROOF. This environment will put the proof in roman type, and will display an end of proof symbol. □

If you want to give a more descriptive title to the proof you can use

```
\begin{proof}[Proof of the existence theorem]
  This environment will put the proof in roman type, and
  will display an end of proof symbol.
\end{proof}
```

which will give

PROOF OF THE EXISTENCE THEOREM. This environment will put the proof in roman type, and will display an end of proof symbol. □

If a proof ends in a display or a list, the end of proof symbol will be placed on the following line. This may leave more space that you want, and can be remedied by inserting a `\qedhere` command. For example,

```
\begin{proof}
  Thus, we conclude that
  \begin{equation*}
    y' = f(t,x).\qedhere
  \end{equation*}
\end{proof}
```

yields

PROOF. ...
Thus, we conclude that

$$y' = f(t,x). \qquad \square$$

Notice that I have used the `equation*` environment instead of double dollar signs for this unnumbered equation. The command `\qedhere` does not work correctly between double dollar signs.

LECTURE 3

The Use of Graphics

There is a large variety of ways to prepare graphics for inclusion in a TEX document. Some of these involve the use of TEX packages, such as `picture`, `pictex` or `Xypic`, which create graphics within TEX itself. Here we are mainly interested in graphics that are prepared using graphics software, and are then imported into a TEX document.

1. AMS restrictions

The AMS puts pretty severe restrictions on the graphics in its publications. For the official information, look at [**AG**]. Here, we will list the three major requirements, and give you some ideas about meeting them.

1.1. Acceptable graphics format

Graphics prepared by separate graphics software must be submitted as encapsulated PostScript (.eps) files.

Of course, most graphics tools have the ability to save graphics in the .eps format. If your documents are in another of the many available graphics formats, you have the choice of redoing them or finding a way to transform them from the original format to .eps.

There are many programs that will convert a graphics file to a different format. However, many of them will only do so with a significant loss of image quality, and not all of these will convert to the .eps format. ImageMagick is available for all operating systems, and works very well. It is freeware and it is available from http://www.imagemagick.org/script/index.php. The program GraphicConverter comes with the standard software package on the Apple.

1.2. Color graphics

The AMS can reproduce color graphics only at additional cost, which must be paid by the author. While one or two color figures can be reproduced at a reasonable cost, a large number can be quite expensive.

If you already have color graphics, you can transform them into acceptable grayscale. Perhaps the best program for doing this is Adobe Illustrator. At full price this is very expensive, but it is possible to get an academic license at a reasonable cost. While Illustrator tends to produce very large files, the results are quite good.

1.3. Line weight

Much mathematical art is line based. The AMS insists that all black lines have a weight (or width) of at least 0.5 point. If you use gray lines, they should have a weight of at least 1.0 point.

This can be a little tricky to interpret. For example, if you produce a graphic containing black lines with a weight of 1 point, but then reduce it to one-third of its origianl size when importing it into your TEX document, it will have a line-weight of one-third point. You have to take your scaling into account when determining the line-weight.

Perhaps the best way to do things is to produce all of your graphics at the size you will use them in your document, and make sure of the line-weights. Otherwise you will have to do the calculation.

If you already have graphics that use line-weights that are too small, you can edit them in Adobe Illustrator. This is often much easier than redoing them.

2. Importing graphics

It is highly recommended that you use the `graphicx` package to import your figures. This will automate many aspects of including graphics, such as numbering, the proper positioning of captions, and to some extent, the proper positioning on the page.

For example, the commands

```
\begin{figure}[h!]
\includegraphics{Fig2.1.4.eps}
\caption{The direction field  for $y’ = y$.}
\label{f2.5}
\end{figure}
```

will produce Figure 1, together with its caption and a label by which it can easily be referenced. (OK, I'm breaking the rules. This one is in color. I didn't want to

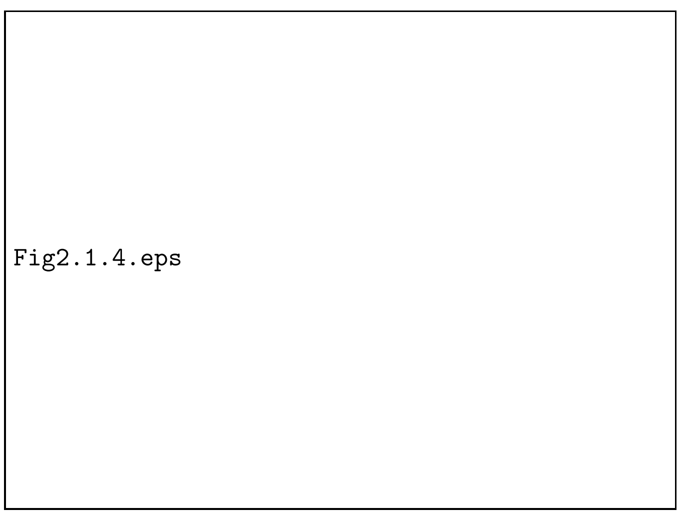

FIGURE 1. The direction field for $y' = y$.

waste time looking for a grayscale figure.)

You can find more information about inserting graphics into TEX documents in [**HK**], p.157 (Section 8.1.3).

LECTURE 4

Resources

This is a list of resources to help you to get started if you are new to TEX, LATEX, or AMS-LATEX, or to find a reference to some advanced topic if you are experienced.

1. Books

If you are a beginner, the AMS suggests [**GR**], *Math into LATEX*, by George Grätzer (http://www.ams.org/bookstore-getitem/item=MLTEX.R). This book is published by Birkhäuser.

A standard reference, useful for both beginners and as a reference for more experienced authors is [**HK**], *A Guide to LATEX*, by Helmut Kopka and Patrick Daly (http://www.ams.org/bookstore-getitem/item=GLTEX). Now in its fourth edition, published by Addison-Wesley, this book includes a discussion of AMS-LATEX.

2. AMS web documents

The AMS offers a wide variety of web documents offering assistance to authors.

- The Author Resource Center
  (http://www.ams.org/authors/)
  This is the main page. It has links to all of those that follow, and more.
- The AMS TEX Resources Home Page
  (http://www.ams.org/tex/)
  A link to information about TEX installation and a source for the AMS-LATEX collection of files.
- Creating graphics for use in AMS books and journals ([**AG**])
  (ftp://ftp.ams.org/pub/author-info/documentation/creating-graphics.pdf)
  This pdf document describes in detail the requirements for graphics in AMS publications. We include this document in our PCMI style package (see Section 3).
- Frequently Asked Questions for AMS Authors
  (http://www.ams.org/authors/author-faq.html)
  An invaluable source of information about almost anything involving TEX and AMS authorship. You can find information about setting up TEX on your own computer, how to put two figures side by side in your document, and much more.
- User's Guide to the `amsmath` package ([**AM**])
  (ftp://ftp.ams.org/pub/tex/doc/amsmath/amsldoc.pdf)
  Provides excellent guidance on the use of math in AMS-LATEX.

- Using the `amsthm` Package ([**AT**])
  (http://www.ams.org/tex/amslatex.html)
  or directly at ftp://ftp.ams.org/pub/tex/doc/amscls/amsthdoc.pdf
  This is the definitive source for the use of the `\theoremstyle` and `\newtheorem` commands discussed briefly in Section 1.2.2. While `\newtheorem` is part of standard LaTeX 2_ε, its use is expanded in $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX. This expansion includes the `\theoremstyle` command as well as new numbering alternatives and a `proof` environment.

3. The PCMI style package

In addition, we provide a number of files. The document style file `pcms-l.cls` is essential. You might find the others useful.

- `pcms-l.cls` — The style file for the preparation of PCMI lecture series.
- *Preparation of PCMI Lecture Notes* — This document, with the document name `pcmistyle.pdf`.
- `pcmistyle.tex` — The TeX source for this document is an example of the use of the document class `pcms-l.cls`.
- Creating graphics for use in AMS books and journals ([**AG**])
  The requirements for graphics in AMS publications.
- `pcms-l-lecture.template` — A template for lecture series using the lectures option. Change the name as appropriate to something ending in .tex and add what you want.
- `pcms-l-article.template` — A template for lecture series using the article option.

Bibliography

[AG] *Creating graphics for use in AMS books and journals,* American Mathematical Society, available online at
ftp://ftp.ams.org/pub/author-info/documentation/creating-graphics.pdf

[AM] *User's Guide to the* `amsmath` *package* American Mathematical Society, available online at ftp://ftp.ams.org/pub/tex/doc/amsmath/amsldoc.pdf

[AT] *Using the* `amsthm` *Package*, American Mathematical Society, available online at
ftp://ftp.ams.org/pub/tex/doc/amscls/amsthdoc.pdf

[GR] George Grätzer, *Math into LaTeX*, 3rd ed., Birkhäuser, Boston (2000)

[HK] H. Kopka & P. W. Daly, *A Guide to LaTeX, Fourth edition*, Addison-Wesley Professional, New York, NY (November 2003)