How to make your own document style in LaTeX2ε

Dag F. Langmyhr

Department of Informatics
University of Oslo
`dag@ifi.uio.no`

1 Background

Both the old LaTeX2.09 and the new LaTeX2ε is distributed with five standard document styles or classes:[1] **article**, **report**, **book**, **slides** and **letter**. Nearly all LaTeX users base their documents on one of these standard classes, perhaps slightly modified by one or more of the myriad of options found on the net. Very few users design new document classes, even though there are several reasons why you might want to do just that:

- The standard classes may not fit your needs. Typical other useful classes would be for lecture notes, technical manuals, theater plays, exam questions, CVs, memos, official letters, etc.
- You are more likely to achieve a good result if you design a complete document class rather than apply several options from various sources.
- You get bored of always seeing the same LaTeX classes.
- It is very easy to design a new document class, particularly since LaTeX2ε came along.

For my own personal use, I needed a class for lecture notes:

- I wanted more space in the margins and between paragraphs, as the students like to make notes.
- I needed an environment for showing short examples of program code, and a 'list of examples' just like the 'list of figures'.
- I wanted a quotation at the start of every chapter.
- I wanted something that did not look quite so 'LaTeXy'.

This class I called **lecnotes**. It was originally designed to work under the old LaTeX2.09, and has since been updated to LaTeX2ε.

2 How to find information

Even though designing a new document class is not difficult, you need some information on how classes work. This information can be found in

- Leslie Lamport: *LaTeX, A Document Preparation System*, Addison-Wesley 1994; second edition, ISBN 0-201-52983-1.
- Michel Goosens, Frank Mittelbach and Alexander Samarin: *The LaTeX Companion*, Addison-Wesley 1994; ISBN 0-201-54199-8.
- The files `clsguide.tex`, which describes how to write a class file, `usrguide.tex`, which introduces LaTeX2.09 users to LaTeX2ε, and `fntguide.tex`, which contains an introduction to the font handling mechanisms of LaTeX2ε. These three files are found in the LaTeX2ε distribution.
- The LaTeX2ε source code itself, in particular the two files `source2e.tex` and `classes.dtx`. These files can be processed by LaTeX2ε to give you an indexed and commented version of the source code.

You also need some proficiency in plain TeX, as the class files are usually written in a mixture of plain TeX and LaTeX.

3 What is a document class?

A document class such as **lecnotes** is really a file `lecnotes.cls` containing plain TeX and LaTeX code. This file is read and processed when its name is found as a parameter to `\documentclass`.

A class file in LaTeX2ε should contain five parts: identification, declaration of options, execution of options, package loading and the main code.

3.1 Writing a class file

When writing a class file for a new document style, you can do this in one of three different ways:

- You can write it from scratch. This is quite difficult and not recommended for beginners.
- You can start with a copy of one of the standard classes and modify it. This is a better approach, but will give you problems if essential parts of LaTeX are changed in a later version.
- You can write a class file which loads a standard class and makes the necessary modifications. This will increase the processing time slightly, but you are much safer against problems with future revisions.
  This is no doubt the preferred way to do it in LaTeX2ε, as a special command `\LoadClass` is available for just this purpose.

Lecnotes was written using `\LoadClass`, and it is based on the standard **report** class.[2]

[1] LaTeX2.09 uses the term 'document style' while LaTeX2 uses 'document class' for the same thing, namely a definition of the appearance of a document; this choice of words is also used in LaTeX2ε to distinguish between the two versions.

[2] The examples shown here contain just the actual code file `lecnotes.cls`. This code is derived from the documented file `lecnotes.dtx`.

3.2 The identification part

This part tells the name of the class and also specifies that it only works under LaTeX2ε:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{lecnotes}[1994/08/26 v1.0
  Private LaTeX2e document class]
```

3.3 The declaration of options part

Some document classes have options and they are declared in this part of the class file. **Lecnotes** has two options, or rather, one option *norsk* for specifying that the text is in Norwegian, and an invalidation of *twocolumn*:[3]

```
\newcommand{\lnotes@exa@id}{Example}
\newcommand{\lnotes@exa@tit}{List
  of Examples}

\DeclareOption{norsk}{%
 \renewcommand{\lnotes@exa@id}{Eksempel}
 \renewcommand{\lnotes@exa@tit}{Eksempler}
}

\DeclareOption{twocolumn}{%
 \ClassWarningNoLine{lecnotes}{option
   'twocolumn' unsupported in this class}}
```

In addition, it has the usual size options *10pt* (see figure 2), *11pt* and *12pt*.

All other options should just be passed on to the **report** class:

```
\DeclareOption*{\PassOptionsToClass
  {\CurrentOption}{report}}
```

3.4 The execution of options part

The previous part just declared the options; now is the time to have them executed. In **lecnotes** I always execute *10pt* in case the user does not specify a size:

```
\ExecuteOptions{10pt}
```

Then all the user-supplied options are executed:

```
\ProcessOptions
```

3.5 The package loading part

As mentioned previously, **lecnotes** is based on the **report** class, so that must be loaded:

```
\LoadClass[twoside]{report}
```

I supply the *twoside* option, as I always use that.

Lecnotes needs two packages (see **Incorporating packages** below) and these must be loaded at this stage:

```
\RequirePackage{float}
\RequirePackage{moreverb}
```

(*Moreverb* also loads the *verbatim* package.)

3.6 The main code part

The rest of this document describes the main code part.

4 Adjusting parameters

There are several style parameters in the standard classes, and part of the work of designing a new class is simply to adjust some of these parameters.

4.1 Adjusting paragraph spacing

The two TeX values `\parindent` and `\parskip` control paragraph indentation and the vertical space added between paragraphs, respectively. Since I wanted no indentation but instead some space between paragraphs, I defined:[4]

```
\setlength{\parskip}{12\p@
      \@plus 4\p@ \@minus 2\p@}
\setlength{\parindent}{\z@}
```

I added some stretch and shrink to the paragraph spacing to avoid vertical spacing problems. The effect of these two definitions is shown in figure 1.

4.2 Adjusting figure placement

Several parameters control how floating material such as figures, tables and examples (see below) is placed. In **lecnotes** I wanted a different layout from the standard one:

- Because of all the program examples I have, I wanted floats to be allowed to fill up to 80% of a text page.
- I wanted up to three floats on a text page, but no bottom floats.
- I wanted separate float pages when necessary, but only when they fill at least 75% of the page.

The specification of this is

```
\setcounter{topnumber}{3}
\renewcommand{\topfraction}{0.80}
\setcounter{bottomnumber}{0}
\setcounter{totalnumber}{3}
\renewcommand{\textfraction}{0.20}
\renewcommand{\floatpagefraction}{0.75}
```

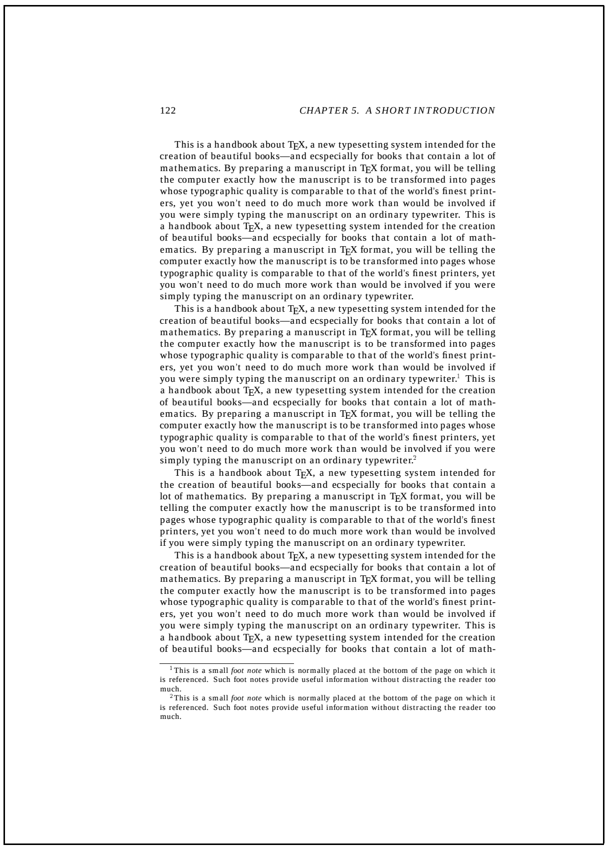
[3]To avoid name conflicts, LaTeX alters the `\catcode` of the '@' character so that it behaves as a letter when reading class and package files.

When writing a class file, it is a good idea to prefix each local name with the name of the class to avoid possible conflict with local names in other class or package files.

[4]To save both storage space and processing time, LaTeX uses commands for words that occur often, such as

`\@depth`	for `depth`
`\@height`	for `height`
`\@minus`	for `minus`
`\p@`	for `pt`
`\@plus`	for `plus`
`\@width`	for `width`

It also uses a dimen register named `\z@` to contain the width 0pt.



122 *CHAPTER 5. A SHORT INTRODUCTION*

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

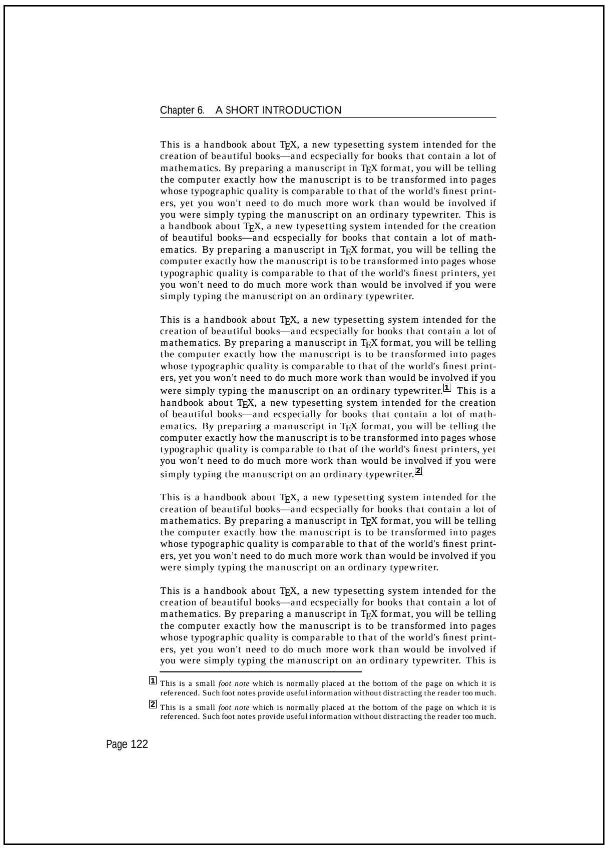
This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.[1] This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.[2]

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of math-

[1] This is a small *foot note* which is normally placed at the bottom of the page on which it is referenced. Such foot notes provide useful information without distracting the reader too much.

[2] This is a small *foot note* which is normally placed at the bottom of the page on which it is referenced. Such foot notes provide useful information without distracting the reader too much.



Chapter 6. A SHORT INTRODUCTION

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.[1] This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.[2]

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TeX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TeX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is

[1] This is a small *foot note* which is normally placed at the bottom of the page on which it is referenced. Such foot notes provide useful information without distracting the reader too much.

[2] This is a small *foot note* which is normally placed at the bottom of the page on which it is referenced. Such foot notes provide useful information without distracting the reader too much.

Page 122

The standard **report** class — The new **lecnotes** class

Figure 1: *A typical text page*

4.3 Roman and Arabic page numbering

In **lecnotes** I wanted the page numbers for the initial part of the document (containing the title page, the abstract, the table of contents, the list of examples, etc.) to be in lower case Roman numbers:

```
\pagenumbering{roman}
```

The first `\chapter` command will change this to Arabic numbers and reset the page counter to 1:

```
    :
  \pagenumbering{arabic}
  \setcounter{page}{1}
    :
```

4.4 Other adjustments

I also made a few other adjustments. I wanted a 'bolder' look, so I made the rules in arrays, tables and framed boxes thicker:

```
\setlength{\fboxsep}{3\p@}
\setlength{\fboxrule}{0.6\p@}
\setlength{\arrayrulewidth}{0.8\p@}
```

5 Choosing a font

A document contains two kinds of text: the body text and the additional text used in headings, headers and footers, figures captions, table of contents, etc. Usually fonts from the same family are used for both kinds of text, but one might achieve a more lively result with two contrasting font families. In **lecnotes** I leave the choice of body text to the writer[5] and use *Gill Sans* in various weights for the additional text.[6]

Since the document class should work for 10, 11 and 12 point body text, I made a macro for each kind of font needed. The fonts defined for the *10pt* option are shown in figure 2.

6 Utilizing hooks

The standard document classes contain many hooks which allow a designer to easily modify the class by redefining simple macroes.

6.1 Designing page styles

LaTeX uses the four macroes `\@evenhead`, `\@oddhead`, `\@evenfoot` and `\@oddfoot` to obtain the page header and footer for a left-hand and right-hand page.[7] A call on a page style such as `\pagestyle{plain}` is just a different notation for calling the macro `\ps@plain` which defines

[5] The example pages shown here are in *New Century Schoolbook*.

[6] In case the user does not have *Gill Sans* available, it is possible to choose another alternative.

[7] If the document does not use the *twoside* option, all pages are regarded as odd-numbered.

```
\newcommand{\lnotes@font@id}{psg}%  % Gill Sans
\newcommand{\lnotes@font}[4]{%
  \fontfamily{\lnotes@font@id}%
  \fontseries{#1}\fontshape{#2}\fontsize{#3}{#4\p@}\selectfont
}

\DeclareOption{10pt}{%
  \newcommand{\lnotes@abs@hed}{\lnotes@font{b}{n}{14}{14}}%  Abstract heading
  \newcommand{\lnotes@cap@num}{\lnotes@font{b}{n}{10}{10}}%  Caption number
  \newcommand{\lnotes@cap@txt}{\lnotes@font{m}{n}{10}{10}}%  Caption text
  \newcommand{\lnotes@cha@app}{\lnotes@font{b}{n}{20}{20}}%  Chapter name
  \newcommand{\lnotes@cha@num}{\lnotes@font{eb}{n}{30}{20}}% Chapter number
  \newcommand{\lnotes@cha@tit}{\lnotes@font{b}{n}{30}{30}}%  Chapter title
  \newcommand{\lnotes@cit@aut}{\lnotes@font{l}{n}{10}{12}}%  Citation author
  \newcommand{\lnotes@cit@txt}{\lnotes@font{l}{it}{10}{12}}% Citation text
  \newcommand{\lnotes@exa@num}{\lnotes@font{m}{n}{7}{0}}%    Example number
  \newcommand{\lnotes@fot@num}{\lnotes@font{b}{n}{8}{0}}%    Footnote number
  \newcommand{\lnotes@hed@sec}{\lnotes@font{b}{n}{17}{18}}%  Section heading
  \newcommand{\lnotes@hed@sse}{\lnotes@font{b}{n}{14}{15}}%  Subsection heading
  \newcommand{\lnotes@hed@sss}{\lnotes@font{b}{n}{12}{12}}%  Subsubsection heading
  \newcommand{\lnotes@hed@par}{\lnotes@font{b}{n}{11}{11}}%  Paragraph heading
  \newcommand{\lnotes@hed@spa}{\lnotes@font{b}{n}{10}{10}}%  Subparagraph heading
  \newcommand{\lnotes@lis@txt}{\lnotes@font{m}{n}{11}{13}}%  List of examples
  \newcommand{\lnotes@num@sec}{\lnotes@font{eb}{n}{14}{14}}% Section number
  \newcommand{\lnotes@num@sse}{\lnotes@font{eb}{n}{12}{12}}% Subsection number
  \newcommand{\lnotes@num@sss}{\lnotes@font{eb}{n}{11}{11}}% Subsubsection number
  \newcommand{\lnotes@num@par}{\lnotes@font{eb}{n}{10}{10}}% Paragraph number
  \newcommand{\lnotes@num@spa}{\lnotes@font{eb}{n}{9}{9}}%   Subparagraph number
  \newcommand{\lnotes@pag@bot}{\lnotes@font{l}{n}{12}{0}}%   Page footer
  \newcommand{\lnotes@pag@top}{\lnotes@font{l}{n}{11}{0}}%   Page header
  \newcommand{\lnotes@tit@pag}{\lnotes@font{b}{n}{10}{10}}%  Title page
}
```

Figure 2: *Various font macros used in the 10pt option*

`\@evenhead` etc.[8] To define a new page style `xxx`, one must supply the `\ps@xxx` macro with the necessary definitions.

Sometimes the header or footer contains the name of the chapter or section found on the page. This is done in three steps:

1. The macro `\chapter` always calls `\chaptermark` with the chapter title as parameter.[9] Similarly, `\section` calls `\sectionmark`, etc.
2. `\chaptermark`, `\sectionmark`, etc., define the appearance of the running title by calling `\markboth` or `\markright`. Each page style defines its own `\chaptermark` and `\sectionmark` (and other `\...mark` if necessary).
3. The running title is then available in `\leftmark` and `\rightmark` for use in `\@evenhead` or one of the others. If there have been several calls on `\markboth` or `\markright` on that page, the first one of these will be given.

I redefined the page style `headings` using the code in figure 3, and the result is shown in figure 1.

Another thing to remember is that some commands like `\chapter` contain a call on `\pagestyle{plain}`. This implies that if one wants to modify the appearance of the first page of a chapter, one must either rewrite `\chapter` or redefine the `plain` page style. I chose the latter approach, and made `plain` a copy of `headings` but with empty headers:

```
\renewcommand{\ps@plain}{\ps@headings
  \renewcommand{\@oddhead}{}%
  \renewcommand{\@evenhead}{}%
}
```

6.2 Modifying the section headings

The main hook for designing section headings is `\@startsection` which has six parameters:

name is the level name for the heading, like `subsection`.

level is the level number for the heading, for instance 2 for a subsection.

indent is the amount of horizontal indentation.

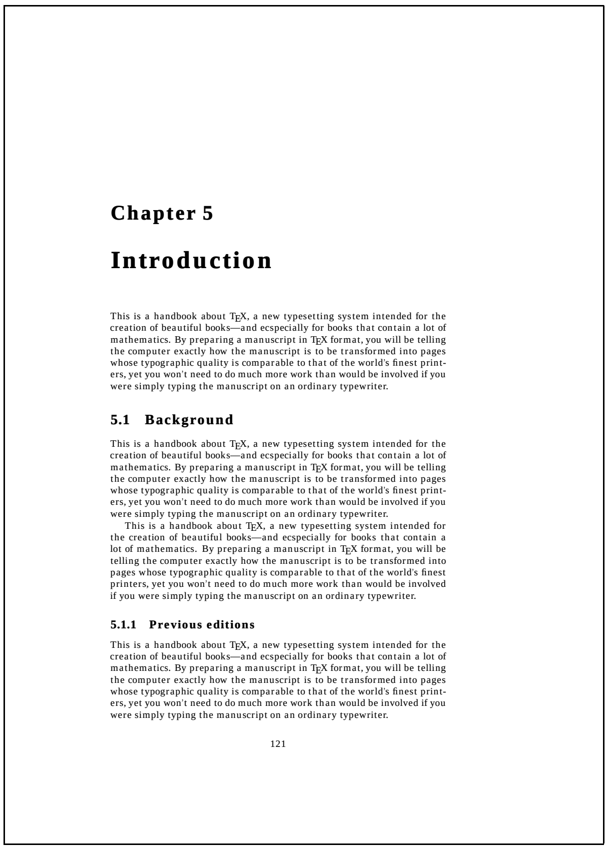
beforeskip is the amount of vertical space to put above each heading.
This parameter serves an additional pupose. If its value is negative, the space above will be the absolute value of the parameter, but in addition the first text paragraph following the heading will not be indented.

afterskip is the amount of vertical space to put below the heading.
This parameter also has a special effect when negative. In this case the heading will be a run-in heading, and the absolute value of the parameter gives the horizontal space between the heading and the following text.

[8] Calling `\thispagestyle` is a little more complex, but that is handled by the LaTeX kernel.
[9] If `\chapter` is used with an optional argument, that optional argument is passed on to `\chaptermark` rather than the main one.

```
\renewcommand{\ps@headings}{%
  \renewcommand{\@oddfoot}{\hfil
    \rlap{\hspace*{1em}\lnotes@pag@bot \pagename\space \thepage}}%
  \renewcommand{\@evenfoot}{%
    \llap{\lnotes@pag@bot \pagename\space \thepage \hspace*{1em}}\hfil}%
  \renewcommand{\@oddhead}{%
    \underline{\raisebox{\z@}[1ex]{%
      \makebox[\textwidth][r]{\lnotes@pag@top \rightmark \strut}}}}%
  \renewcommand{\@evenhead}{%
    \underline{\raisebox{\z@}[1ex]{%
      \makebox[\textwidth][l]{\lnotes@pag@top \leftmark \strut}}}}%
  \def \chaptermark ##1{%
    \markboth{\@chapapp\space \thechapter.\space\space\space \uppercase{##1}}%
             {}}%
  \def \sectionmark ##1{%
    \markright{\thesection.\space\space\space \uppercase{##1}}}%
}% end of \ps@headings
```

Figure 3: *Defining the page style headings.*



Chapter 5

Introduction

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

5.1 Background

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

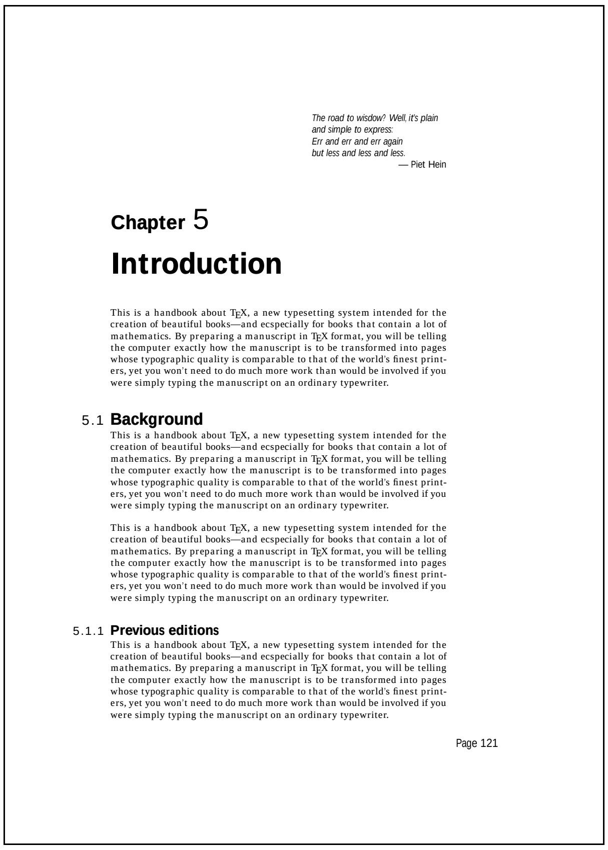
This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

5.1.1 Previous editions

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

121

The standard **report** class



The road to wisdow? Well, it's plain
and simple to express:
Err and err and err again
but less and less and less.
— Piet Hein

Chapter 5

Introduction

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

5.1 **Background**

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

5.1.1 **Previous editions**

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

Page 121

The new **lecnotes** class

Figure 4: *Chapter and section headings*

style specifies which type style to use when printing the heading, for example `\Large\bf`.

Using this hook it is easy to define the various section headings. My definitions differ from the standard ones in both spacing and font style and are shown in figure 4:[10]

```
\renewcommand{\section}{%
  \@startsection{section}{1}{\z@}%
  {2.5ex \@plus 1ex \@minus 0.2ex}%
  {-0.75\baselineskip}{\lnotes@hed@sec}}
\renewcommand{\subsection}{%
  \@startsection{subsection}{2}{\z@}%
  {2.00ex \@plus 0.8ex \@minus 0.2ex}%
  {-0.80\baselineskip}{\lnotes@hed@sse}}
\renewcommand{\subsubsection}{%
  \@startsection{subsubsection}{3}{\z@}%
  {1.50ex \@plus 0.6ex \@minus 0.2ex}%
  {-0.85\baselineskip}{\lnotes@hed@sss}}
\renewcommand{\paragraph}{%
```

[10]The sign of the fourth and fifth parameters may not be as you expected, but that is because I have redefined `\@startsection`; see the section **Modifying the code** below.

```
  \@startsection{paragraph}{4}{\z@}%
  {-1.0ex \@plus -0.4ex \@minus -0.2ex}%
  {1em}{\lnotes@hed@par}}
\renewcommand{\subparagraph}{%
  \@startsection{subparagraph}{5}{\z@}%
  {-0.01ex \@plus -0.2ex \@minus -0.01ex}%
  {1em}{\lnotes@hed@spa}}
```

The appearance of the section number is defined in a hook of its own called `\@seccntformat` having the section level name as its parameter. I redefined that to put the numbers into the margin (using `\llap`) and in a font of their own:

```
\renewcommand{\@seccntformat}[1]{\llap
  {\csname lnotes@#1\endcsname
    \hspace*{6\p@}}}
\newcommand{\lnotes@section}{%
  {\lnotes@num@sec \thesection}}
\newcommand{\lnotes@subsection}{%
  {\lnotes@num@sse \thesubsection}}
\newcommand{\lnotes@subsubsection}{%
  {\lnotes@num@sss \thesubsubsection}}
\newcommand{\lnotes@paragraph}{%
  {\lnotes@num@par \theparagraph}}
\newcommand{\lnotes@subparagraph}{%
  {\lnotes@num@spa \thesubparagraph}}
```

6.3 Modifying the chapter and part headings

The chapter and part headings are more complex and usually not made with `\@startsection`. Chapter headings are defined by the macro `\@makechapterhead` (or `\@makeschapterhead` if `\chapter*` was used) and part headings by `\@part` (or `\@spart`), and these can be modified if needed.

I redefined `\@makechapterhead` to use different spacing and font, and also to include a short quotation defined previously using a new command called `\chaptercite`; the result can be seen in figure 4.

6.4 Modifying the table of contents

There exist hooks for designing the appearance of each line in the table of contents, and these hooks are called `\l@part`, `\l@chapter`, etc. There also exist two hooks named `\l@figure` and `\l@table` for defining the entries in the lists of figures and tables.

Each line on the table of contents file `file.toc` is on the form

```
\contentsline{section}%
  {\numberline{2.1}Hex numbers}{14}
```

indicating that *section* number *2.1* titled *Hex numbers* starts on page *14*. This is just a different notation for calling the right hook:

```
\l@section
  {\numberline{2.1}Hex numbers}{14}
```

Since most tables of contents look rather alike, with a dotted line leading from the section title to the page number, LaTeX provides a macro to make such a line. It is called `\@dottedtocline` and has five parameters:

level is the section level number, for instance 1 for a section.

indent is the amount of indentation to the left.

numwidth is the width to be reserved for the section number.

text is the section title. This usually consists of two parts: the section number, given as a parameter to `\numberline`, and the title text. The macro `\numberline` just sets its parameter left justified in a box of the width specified by the third parameter to `\@dottedtocline`.

page is the page number. This is set right justified in a box of width `\@pnumwidth`, which may be redefined by the user.

It is now easy to define the three hooks `\l@section`, `\l@subsection` and `\l@subsubsection` using `\@dottedtocline`:[11]

```
\renewcommand{\l@section}{%
  \@doubledottedtocline{1}{2.5em}{2em}}
\renewcommand{\l@subsection}{%
  \@doubledottedtocline{2}{4.5em}{3em}}
\renewcommand{\l@subsubsection}{%
  \@doubledottedtocline{3}{7.5em}{4em}}
```

(Since the two parameters to `\l@section` etc. are to be transmitted unaltered to `\@dottedtocline`, they have been omitted in these definitions.)

6.5 Modifying the footnotes

In **lecnotes** I wanted the footnote markers to be more striking than in the standard classes, so I redefined `\@makefnmark` to place them in a box as shown in figure 1:

```
\renewcommand{\@makefnmark}{%
  \hbox{\hspace*{0.1em}%
    \setlength{\fboxsep}{1.0\p@}%
    \setlength{\fboxrule}{0.4\p@}%
    \raisebox{0.8ex}{\fbox{%
      \lnotes@fot@num \@thefnmark}}%
    \hspace*{0.1em}}}%
```

I also redefined `\@makefntext` to change the appearance of the actual footnote; in particular I wanted the footnote marker to ‘hang out’ in the left margin, and I wanted a wider paragraph indentation:

```
\long\def \@makefntext #1{%
  \setlength{\parindent}{3em}\noindent
  \llap{\@makefnmark \hspace*{3\p@}}#1}
```

I wanted more space between the footnotes:

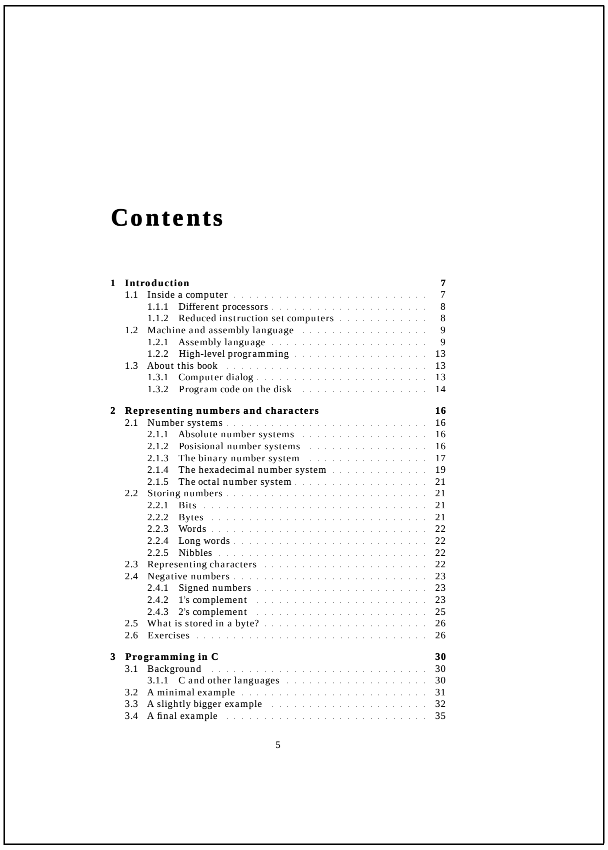
```
\setlength{\footnotesep}{12\p@}
```

Finally, I wanted a longer and thicker rule separating the footnotes from body text:[12]

```
\renewcommand{\footnoterule}{%
  \vspace*{-3\p@}
  \hrule \@width 0.6\columnwidth
        \@height 0.8\p@ \relax
  \vspace*{2.2\p@}}
```

[11] In **lecnotes** I use `\@doubledottedtocline` instead. It is my own very simple modification of `\@dottedtocline` placing the dots in groups of two, as shown in figure 5.

[12] LaTeX requires that `\footnoterule` occupies no vertical space, so it must contain negative spacing equal to the thickness of the rule and the positive spacing.



Contents

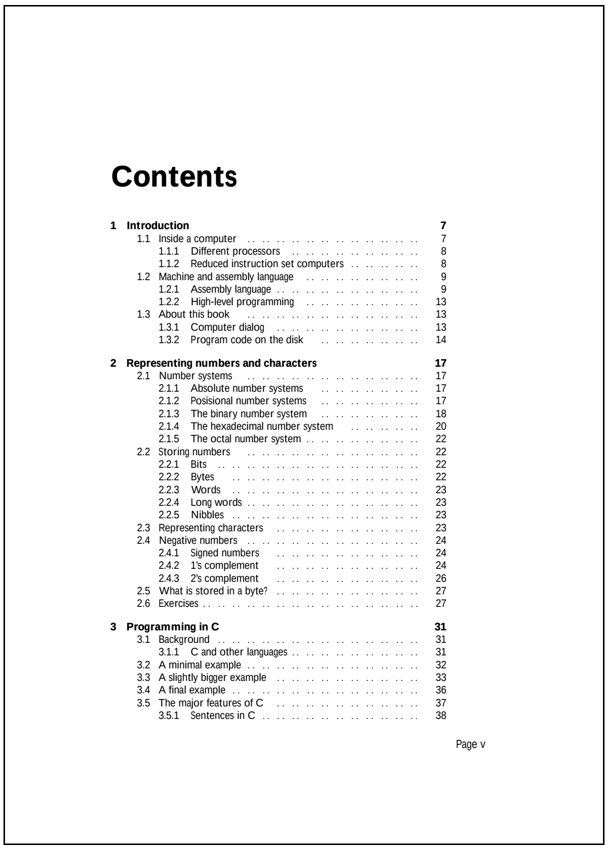
1 Introduction **7**
1.1 Inside a computer 7
1.1.1 Different processors 8
1.1.2 Reduced instruction set computers 8
1.2 Machine and assembly language 9
1.2.1 Assembly language 9
1.2.2 High-level programming 13
1.3 About this book 13
1.3.1 Computer dialog 13
1.3.2 Program code on the disk 14

2 Representing numbers and characters **16**
2.1 Number systems 16
2.1.1 Absolute number systems 16
2.1.2 Posisional number systems 16
2.1.3 The binary number system 17
2.1.4 The hexadecimal number system 19
2.1.5 The octal number system 21
2.2 Storing numbers 21
2.2.1 Bits 21
2.2.2 Bytes 21
2.2.3 Words 22
2.2.4 Long words 22
2.2.5 Nibbles 22
2.3 Representing characters 22
2.4 Negative numbers 23
2.4.1 Signed numbers 23
2.4.2 1's complement 23
2.4.3 2's complement 25
2.5 What is stored in a byte? 26
2.6 Exercises 26

3 Programming in C **30**
3.1 Background 30
3.1.1 C and other languages 30
3.2 A minimal example 31
3.3 A slightly bigger example 32
3.4 A final example 35

5

The standard **report** class



Contents

1 Introduction **7**
1.1 Inside a computer 7
1.1.1 Different processors 8
1.1.2 Reduced instruction set computers 8
1.2 Machine and assembly language 9
1.2.1 Assembly language 9
1.2.2 High-level programming 13
1.3 About this book 13
1.3.1 Computer dialog 13
1.3.2 Program code on the disk 14

2 Representing numbers and characters **17**
2.1 Number systems 17
2.1.1 Absolute number systems 17
2.1.2 Posisional number systems 17
2.1.3 The binary number system 18
2.1.4 The hexadecimal number system 20
2.1.5 The octal number system 22
2.2 Storing numbers 22
2.2.1 Bits 22
2.2.2 Bytes 22
2.2.3 Words 23
2.2.4 Long words 23
2.2.5 Nibbles 23
2.3 Representing characters 23
2.4 Negative numbers 24
2.4.1 Signed numbers 24
2.4.2 1's complement 24
2.4.3 2's complement 26
2.5 What is stored in a byte? 27
2.6 Exercises 27

3 Programming in C **31**
3.1 Background 31
3.1.1 C and other languages 31
3.2 A minimal example 32
3.3 A slightly bigger example 33
3.4 A final example 36
3.5 The major features of C 37
3.5.1 Sentences in C 38

Page v

The new **lecnotes** class

Figure 5: *The table of contents*

7 Incorporating packages

There is little sense in reinventing features implemented by others, so when writing a document class some of the work is to make those features available, perhaps with a different interface.

To create a suitable command for showing program examples, as shown on figure 6, I found the material I needed in three packages:

Verbatim by Rainer Schöpf makes it possible to include verbatim code from a file.

Moreverb by Angus Duggan extends *verbatim* with line numbers and correct handling of TAB characters.

Float by Anselm Lingnau makes it possible to design new kinds of floats.

The loading of these packages was specified in the package loading part of the class file.

I could now create the new float environment I needed:

```
\newfloat{example}{tp}{loe}[chapter]
\floatname{example}{\lnotes@exa@id}
\floatstyle{plain}
```

Using this, I could make a command `\examplefile` with two parameters: the name of the example file, and a caption:

```
\newcommand{\examplefile}[2]{%
  \begin{example}
    %% First the top rules:
    \hrule \@height 1.0\p@ \vspace{2\p@}
    \hrule \@height 0.4\p@ \vspace{4\p@}
    %% Then the example text:
    {\renewcommand{\baselinestretch}{0.85}
      \listinginput{1}{#1}}
    %% After that, the bottom rules:
    \vspace{4\p@}\hrule \@height 0.4\p@
    \vspace{2\p@}\hrule \@height 1.0\p@
    \relax
    %% And finally, the caption:
    \caption{#2}
  \end{example}}
```

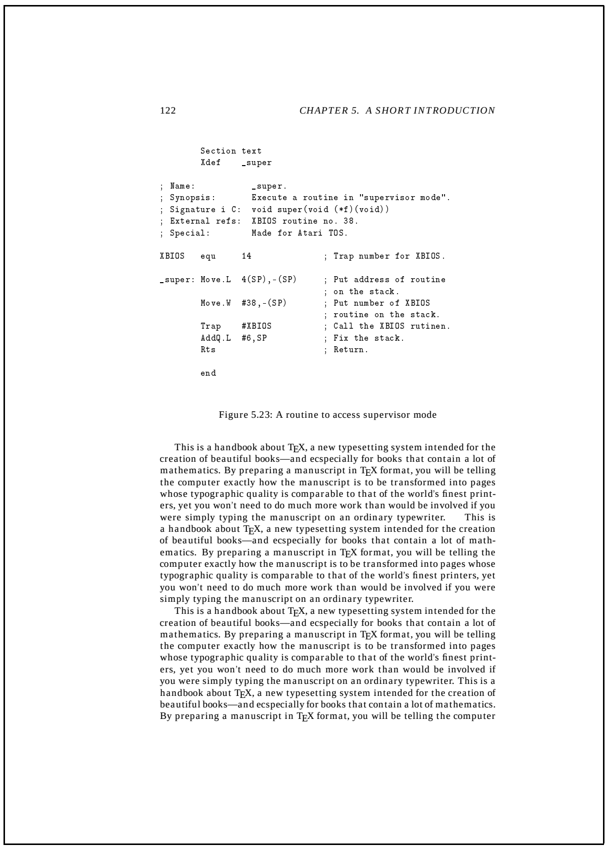
I added the two rules above and below the code to make it stand out more clearly from the body text.

The final touch is the macro `\listofexamples`. All the necessary code was in the *float* package; I only needed to add some font specifications:

```
\newcommand{\listofexamples}{%
  {\lnotes@lis@txt
    \setlength{\parskip}{\z@ \@plus 1\p@}%
    \renewcommand{\rmdefault}{%
      \lnotes@font@id}%
    \rmfamily
    \listof{example}{\lnotes@exa@tit}%
}}
```

8 Modifying the code

Even though you can get quite a long way just using the parameters and hooks supplied by the LaTeX designers, you sometimes need to modify the actual LaTeX code.



122 *CHAPTER 5. A SHORT INTRODUCTION*

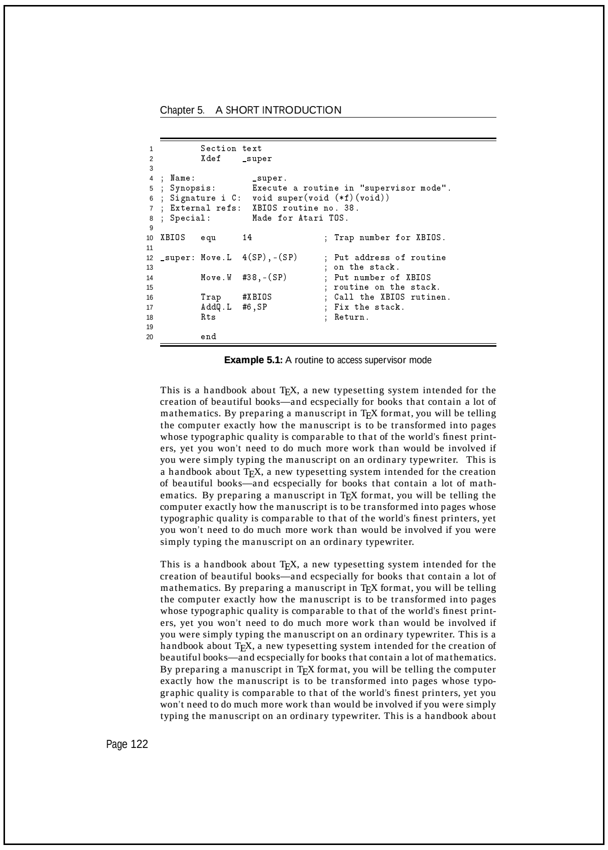
```
        Section text
        Xdef    _super

; Name:           _super.
; Synopsis:       Execute a routine in "supervisor mode".
; Signature i C:  void super(void (*f)(void))
; External refs:  XBIOS routine no. 38.
; Special:        Made for Atari TOS.

XBIOS   equ     14              ; Trap number for XBIOS.

_super: Move.L  4(SP),-(SP)     ; Put address of routine
                                ; on the stack.
        Move.W  #38,-(SP)       ; Put number of XBIOS
                                ; routine on the stack.
        Trap    #XBIOS          ; Call the XBIOS rutinen.
        AddQ.L  #6,SP           ; Fix the stack.
        Rts                     ; Return.

        end
```

Figure 5.23: A routine to access supervisor mode

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer

The standard **report** class



Chapter 5. A SHORT INTRODUCTION

```
        Section text
        Xdef    _super

; Name:           _super.
; Synopsis:       Execute a routine in "supervisor mode".
; Signature i C:  void super(void (*f)(void))
; External refs:  XBIOS routine no. 38.
; Special:        Made for Atari TOS.

XBIOS   equ     14              ; Trap number for XBIOS.

_super: Move.L  4(SP),-(SP)     ; Put address of routine
                                ; on the stack.
        Move.W  #38,-(SP)       ; Put number of XBIOS
                                ; routine on the stack.
        Trap    #XBIOS          ; Call the XBIOS rutinen.
        AddQ.L  #6,SP           ; Fix the stack.
        Rts                     ; Return.

        end
```

Example 5.1: A routine to access supervisor mode

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter.

This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about TEX, a new typesetting system intended for the creation of beautiful books—and ecspecially for books that contain a lot of mathematics. By preparing a manuscript in TEX format, you will be telling the computer exactly how the manuscript is to be transformed into pages whose typographic quality is comparable to that of the world's finest printers, yet you won't need to do much more work than would be involved if you were simply typing the manuscript on an ordinary typewriter. This is a handbook about

Page 122

The new **lecnotes** class

Figure 6: *A program code example*

One example is the `\@startsection` command which does not allow negative space after a heading. Since I use `\parskip`$>$0, I get too much vertical space between the heading and the following text. The only solution is to redefine `\@startsection` to use the fourth parameter to specify display or run-in heading rather than the fifth. No lines are indented in the **lecnotes** class anyway.

8.1 Redefining the abstract environment

Another example is the *abstract* environment which I want to start on a odd-numbered page, use a special font, and produce an extra blank page if *twoside* is used. The only way to achieve this, is to redefine it:

```
\renewenvironment {abstract}{%
    \cleardoublepage
    \begin{titlepage}
      \mbox{}\vfill
      \centerline{\lnotes@abs@hed
        \abstractname}%
  }{%
      \par\vfill
      \mbox{}%
    \end{titlepage}
    \if@twoside
      \begin{titlepage}\mbox{}
      \end{titlepage}
    \fi
  }
```

8.2 Modifying the theindex environment

One does not always have to redefine the whole piece of code, since it is easy to keep the original under a different name using a `\let`, and just specify some additional action. For instance, to ensure that the *theindex* environment always starts on an odd-numbered page, the following will suffice:[13]

```
\let \lnotes@theindex = \theindex
\let \lnotes@endtheindex = \endtheindex

\renewenvironment{index}%
  {\cleardoublepage \lnotes@theindex}%
  {\lnotes@endtheindex}
```

9 Conclusion

My intention with this description of **lecnotes** is not to promote it as a great new document class, which it is not, but to show that it is quite easy to create a new document class. My hope is that others will make classes of their own, and make them widely available.

The **lecnotes** class can be found at `ftp.ifi.uio.no: pub\tex\lecnotes.cls`.

[13] Internally, LaTeX implements a `\begin{xxx}` as a `\xxx`, and an `\end{xxx}` as an `\endxxx`.