

前言

1978 年美国著名数学家与计算机专家，斯坦福大学的 Donald E. Knuth 教授发明了 TeX 排版系统，其精美的排版效果立即引起学术界和出版界的一片赞叹，美国数学学会率先采用。1985 年美国数学家与计算机专家 Leslie Lamport 博士，在 TeX 的基础上开发出更便于普通用户掌握的 LaTeX 排版系统。1989 年以德国数学家 Frank Mittelbach 为首组成 LaTeX3 项目小组，负责对 LaTeX 的维护和开发工作，并于 1994 年推出目前广为使用的版本 LaTeX2e。

LaTeX 的缺点就是文稿编排不直观，命令繁杂，不易短时间熟练掌握。为此，国内外许多大学、出版公司和学术会议都提供相关的 LaTeX 模板，以便于论文写作。尽管如此，要想顺利完成论文写作，还是需要对 LaTeX 有所了解，如表格编排、插图处理等；而且各种模板互不兼容，如果不熟悉 LaTeX，就很难灵活使用和修改；也就是说，学会使用某个 LaTeX 模板并不等于就会使用其他模板，更不等于学会使用 LaTeX 写作。

编著本书的目的是希望：初学者读后可以无师自通，能够自行顺利地使用 LaTeX 完成论文写作；对于熟悉 LaTeX 的读者，可作为这方面的工具书，随时查阅，例如遇到表格问题，可查阅**表格一章**，若是编译出现问题，可查阅**编译一章**，如果需要制作陈述幻灯片，可参阅**幻灯片一章**。

近年来国内使用最为广泛的中文 LaTeX 系统是 CTeX 中文套装，它所附带的文本编辑器是 WinEdt，本书就是使用 CTeX 在 WinEdt 中编写的，书中介绍的所有 LaTeX 写作方法、技巧和示例都通过了 CTeX 的编译检查。

本书自 2011 年 1 月出版以来收到大量热心读者的来信，他们提出了很多有益的建议和希望，与此同时 CTeX 和 WinEdt 的版本也多次更新升级，这些是修订本书的根本原因。第二版除了修正初版的几处错误外，主要是更新对 CTeX 和 WinEdt 的说明，添加了对 XeLaTeX 和很多实用宏包的介绍，示例也从 303 个增加到 512 个。

本书附带一张 DVD 资料光盘，其中的资料由两部分组成：1. CTeX 中文套装，它分为 Basic 和 Full 两种版本。2. 本书全部示例的源文件及其编译后生成的 PDF 文件。

本书在编写过程中参考了国内外许多学者的相关论著，在此向他们表示感谢和敬意。还要感谢 CTEX 网站向社会无偿提供 CTeX 中文套装，感谢 CTEX 网站站长、中科院应用数学所副研究员吴凌云博士应邀为本书作序。感谢香港中文大学胡海博士对书稿的校对和修改意见，感谢清华大学出版社夏兆彦、赖晓等编辑人员的大力支持和帮助。

LaTeX 博大精深，文献资料浩如烟海，编者学识有限，书中难免有错误和欠缺之处，敬请批评指正或提出修改建议；所发现的错误及其更正将随时公布在 <http://zzg34b.w3.c361.com/errata.htm> 网页。

感谢您阅读本书，并期待您的宝贵意见。

编者
latexer2010@gmail.com
2012 年 11 月

第 2 章 LaTeX 基础

本章所介绍的内容都是在使用 LaTeX 写作时所需掌握的最基本的知识。在论文写作过程中，所出现的问题究其原因很多都是对这些基础知识没有正确理解。磨刀不误砍柴工，为了能够熟练和灵活地运用 LaTeX 进行写作，尽量减少错误的出现，就必须耐心细致地研习这些基础知识，这将会收到事半功倍的效果。

2.1 源文件的结构

所有 LaTeX 源文件都可分为导言和正文两大部分；由于短篇论文与中长篇论文的篇幅和所采用的层次结构不同，它们在正文部分的结构有较大区别。

2.1.1 短篇论文

通常在 10 页以内的、不设置目录的论文称为短篇论文，它们大都是由若干个节和小节组成的，其源文件的基本结构如图 2.1 所示。

```
\documentclass{article} % 使用 article 文档类型格式排版
\usepackage{amsmath}    % 调用公式宏包
\usepackage{graphicx}   % 调用插图宏包
\begin{document}
论文内容
\end{document}
```

图 2.1 短篇论文源文件的基本结构

从源文件的第一行命令 `\documentclass` 开始，到命令 `\begin{document}` 之前的所有命令语句统称为导言；在 `\begin{document}` 与 `\end{document}` 之间的所有命令语句和文本统称为正文。命令 `\end{document}` 之后的任何字符，LaTeX 都将忽略。

导言主要由文档类型命令和调用宏包命令以及其他设置命令组成，其作用是对文稿的排版格式和排版功能进行设置，在导言中的任何设置都将对正文的全文产生影响。

正文是由论文内容包括文本、插图、表格、公式等和各种 LaTeX 命令组成，正文中的命令只对其后的局部正文产生影响。

短篇论文通常使用 `article` 文类或出版机构提供的专用文类。由于短篇论文篇幅短小，层次结构简单，直接将论文内容写入正文区域就可以了。

图 2.1 中的 `%` 符号是注释符，它表示其右侧的文字是对左侧命令或文本的说明；在编译源文件时，LaTeX 将忽略注释符及其右侧的所有字符。

2.1.2 中长篇论文

一般篇幅在 10 页以上至几百页的、需要设置目录的论文称为中长篇论文，它们都是由若干个章组成的，其源文件的典型结构如图 2.2 所示。

中长篇论文通常使用 `book` 文类、`report` 文类或出版机构提供的专用文类。

第3章 字体

在 LaTeX 中，字体分为文本字体和数学字体两类，通常文本字体只能在文本模式中使用，数学字体只能在数学模式中使用。本章主要介绍文本字体的选择和使用，数学字体将在 **数学式** 一章中专门介绍。

传统的字体选择方式是用指定字体的属性来确定的，作者无需了解字体的名称和访问的路径，但只限于 CTeX 系统之内的字体。自从出现 XeLaTeX 之后，字体还可以直接使用字体名或字体文件名来确定，而且它适用于计算机内的任何 TrueType 或 OpenType 字体，只要知道它们的存取路径就行。

3.1 字体的属性

在 LaTeX 中，每种文本字体都具有五个属性，它们各自独立而又相互依存，即改变字体的某一属性，并不影响其他属性。

3.1.1 字体的五个属性

字体的五个属性及其说明如表 3.1 所示。

表 3.1 字体的五个属性及其说明

属性	说明
encoding	编码，将字符按照某种规定的方式编制成相应的代码信息，例如 OT1、T1 编码
family	字族，某一类型字体集合的名称，例如 Computer Modern Roman 字族、仿宋字族和方正舒体字族等。字族就是通常所说的字体
series	序列，字体笔画的粗细和宽窄程度
shape	形状，字体的外表形态，如倾斜、直立等
size	尺寸，字体的大小，以点数表示，如 11pt、20pt 等

3.1.2 属性的默认值和可选值

字体的每个属性都有多个可选值，只有分别确定每个属性的可选值，字体才能最终被确定。所以要了解每个属性及其各种选项分别对输出字体的影响和作用。

(1) 在使用 LaTeX 写论文时会遇到两种编码：输入字体编码和输出字体编码。输入字体编码就是通过键盘或者复制粘贴输入到文本编辑器的字符编码，在 WinEdt 中所支持的输入字体编码主要有 UTF-8 和 ANSI 两种，这已经在 1.5.2 安装与测试中说明。本章中的字体编码是指输出字体编码，就是经编译输出到 PDF 文件中的字符编码。为了能够在字体文件中查找所需字符，就要对其中所有字符进行编码。字体文件，俗称字库，是某一类字形和符号的显示描述文件；编码就是在字体文件中查找字符的索引表。LaTeX 采用的字符编码如下。

- 文本字体的默认编码是 OT1，也可使用字体属性命令改为 T1 编码或者其他编码。OT1 编码的字符只有 128 个，其中包含英文字母、阿拉伯数字和标点符号，如果需要用到西欧文字中的变音字母，例如法文中的 é，只能用符号命令生成；而 T1 编码的字符有 256 个，它在 OT1 字符的基础上增加了 128 个西欧文字符号。

第 4 章 版面设计

如果说 TeX 是优秀的排字工，那 LaTeX 就是杰出的版面设计师。其实只要在源文件的第一条命令中选定所需的文类，版面设计工作就结束了，系统会自动按照最典型的英文论文的版面格式来设置版面。但由于各院校和出版机构都有自己的版面格式要求，每个人的审美取向也各不相同，一种版面格式再完美也满足不了所有需求，于是 LaTeX 还提供了一系列用来修改版面格式的版面设置命令，使用相关的宏包也可以很方便地进行版面设计。

4.1 版面

版面就是书刊每一页的整面，由版心、页眉、页脚和边注以及四周空白构成，它们被称为版面元素。

4.1.1 版面元素的位置

各种版面元素的位置及其尺寸如图 4.1 所示。版心位于版面的中央，论文的所有内容除边注外都排版于版心中；页眉和页脚分别位于版心的上方和下方，主要用来排版当前页所属

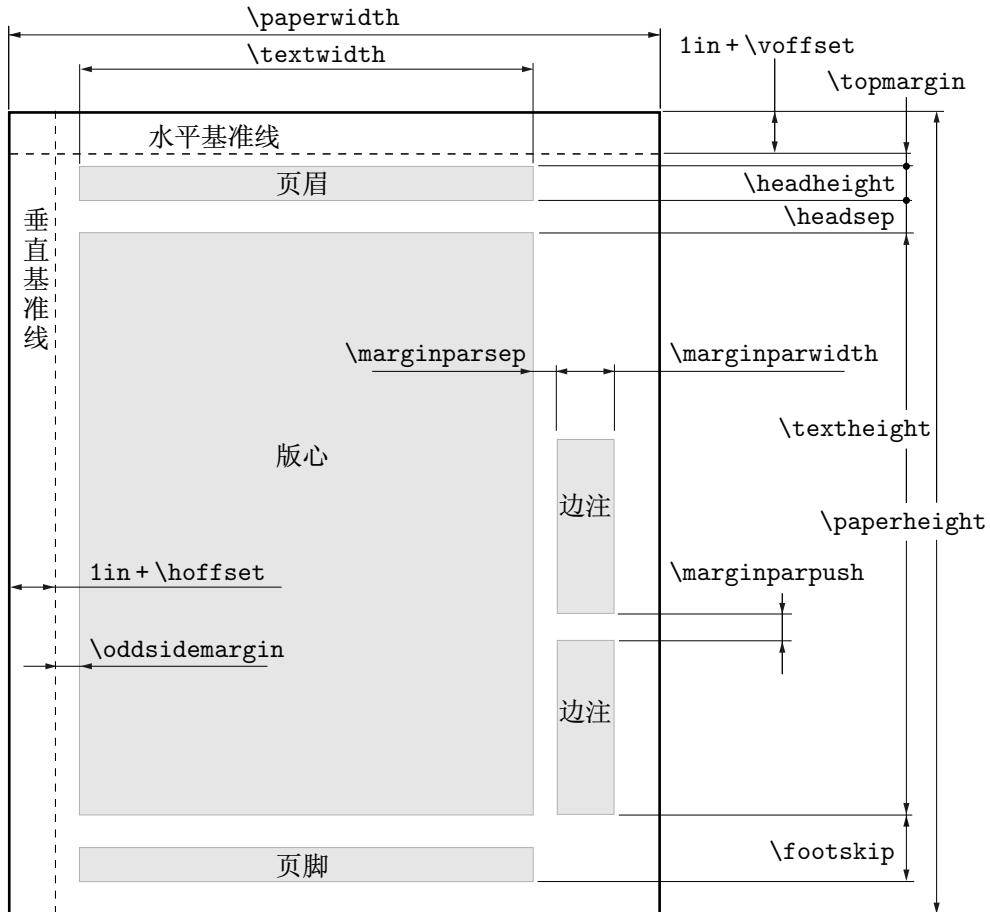


图 4.1 版面元素的位置及其尺寸示意图

章节的标题和页码，以便于读者对论文的阅读和检索。

第 8 章 数学式

在科技论文中经常要用到分式、根式、行列式和微积分方程式以及证明、定理等数学表达式，本书将它们统称为数学式。TeX 就是因数学式而生，LaTeX 就是为数学式而长，排版数学式乃是 LaTeX 的最强项，而且是数学式越复杂越能显示出它的优越性，再辅以相关的数学宏包，尤其是 `amsmath` 公式宏包，将大幅度地扩充 LaTeX 的数学式排版功能，使其排版效果更为精美和专业。

8.1 数学模式

在 LaTeX 中，最常用到的主要有文本模式和数学模式这两种模式。数学模式又可分为行内公式 (inline math) 和行间公式 (display math) 两种形式。

行内公式形式是将数学式插入文本行之内，使之与文本融为一体，这种形式适合编写简短的数学式。

行间公式形式是将数学式插在文本行之间，自成一行或一个段落，与上下文附加一段垂直空白，使数学式突出醒目。多行公式、公式组和微积分方程等复杂的数学式都是采用行间公式形式编写。

8.1.1 行内公式

LaTeX 提供了以下三种方法来编写行内公式。

- (1) \$... \$;
- (2) \(...\);
- (3) \begin{math} ... \end{math}。

例 8.1 同一个行内公式，分别使用三种方法编写，比较三者的排版效果。

抛物线 $y^2=2px$ 的切线方程 \\
 抛物线 $\backslash(y^2=2px)$ 的切线方程 \\
 抛物线 $\backslash\begin{math} y^2=2px \end{math}$ 的
切线方程

抛物线 $y^2 = 2px$ 的切线方程
 抛物线 $y^2 = 2px$ 的切线方程
 抛物线 $y^2 = 2px$ 的切线方程

可以看出，三种编写方法的排版结果是完全相同的，不过第一种方法比较常用，因为它性格坚强（见本书编译一章中的说明），使用方便，其缺点是起止符同为 \$，不能区分起止，常被遗漏，从而造成编译出错。

这三种行内公式的编写方法既可用于段落模式也可用于左右模式。段落中的行内公式应简短，以避免中断换行，因为系统只能在某些数学符号处自动换行。

例 8.2 如果行内公式较长，系统会在 = < > 等关系符和 + - 等二元符处自动换行。

函数 $f(x)=a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}$

函数 $f(x) = a_nx^n + a_{n-1}x^{n-1} +$
 $a_{n-2}x^{n-2}$

若禁止在关系符或二元符处换行，可使用 `\relpenalty=10000` 或 `\binoppenalty=10000` 赋值命令。这两个控制命令的默认值分别是 500 和 700，其取值范围都是 $-10000 \sim 10000$ ，它代表从必须到禁止的坚决程度。也可在禁止换行处直接插入 `\nobreak` 命令。

第9章 插图

西方学者常说：A picture is worth a thousand words，据说是源自中国的成语：百闻不如一见，把它译为一图胜千言，非常准确地表达了这个成语最初的意思，就是说，一幅图形可以简单明确地表达很多错综复杂、千言万语都难以描述的事物信息。当然有些精辟的文字也是图形难以替代的，例如：“大漠孤烟直，长河落日圆”，那给人以苍凉、壮美和思乡的感受就无法用图形来描绘。所以，一篇优秀的论文应该是图文并茂，相得益彰。

论文中的插图既是研究成果真实准确的表述，也是作者计算机绘图水平和插图技巧的展示。LaTeX 的绘图功能比较简单，如果使用与 LaTeX 相关的各种绘图宏包或绘图工具，例如 pgf、pstricks、tikz 和 METAPOST 等，也可以画出非常复杂的图形，但缺点是作图不直观，绘画命令繁多，不易熟练掌握。现在，通常是用 Excel、Matlab 和 Visio 等功能强大的可视绘图工具先把图形画好，例如条形图、流程图和曲线图等，然后插入到 LaTeX 源文件中。所以本书不再详细介绍如何用 LaTeX 绘图，而着重于如何插图到 LaTeX 源文件中。如果有读者希望了解 LaTeX 绘图技巧，可查阅相关文献 [20]。

9.1 图形的种类

现在，图形的存储格式很多，一般来讲可将它们分为两大类：位图图形和向量图形。无论哪种图形都是以数字形式存储，只是它们的解释方法各不相同。

9.1.1 位图图形

位图图形，也称点阵图形，在技术上称为栅格图像，它使用称作像素的小方形组成的网格来表示图像。每个像素都有自己特定的位置和颜色值。处理位图图像时，所编辑的是像素，而不是对象或形状。位图图形是连续色调图像（如照片或数字绘画）最常用的图形文件格式，因为它可以表现阴影和颜色的细微层次。位图图形与分辨率有关，也就是单位长度中像素的数量。一幅位图图形确定后，其像素数量固定不变，无论放大还是缩小只不过是改变每个像素的尺寸大小。因此，如果在屏幕上对位图图形进行缩放、旋转或以低于创建时的分辨率来打印，都会丢失其中的细节，或出现锯齿现象，造成失真。

分辨率越高，图形的清晰度就越高；像素的颜色值（色深）越多，图形的色彩就越逼真，如果色深为 8bit 只有 256 种颜色，而 24bit 色深可得到 1600 万种颜色，但图形文件的尺寸也会随之急剧增大。

位图图形还可分为无损压缩格式和有损压缩格式。无损压缩的优点是能够完好地保存图形的像素信息，但是压缩率比较低。有损压缩技术可以大幅度地压缩图形文件，但会使图形的质量降低。常用的图形文件中 TIFF、PNG 和 GIF 是无损压缩格式，JPG 是有损压缩格式。

9.1.2 向量图形

向量图形是由用数学公式定义的线段和曲线组成的图形，这些线段和曲线称为向量。改变这些向量的位置、形状、长短或颜色都不会影响图形的品质。向量图形与分辨率无关，也就是说，可以将其任意缩放或旋转，按任意分辨率打印，都不会失真。因此，向量图形最适合表现醒目的图形。

第 10 章 正文工具

本章提到的正文是指论文中的正文部分，通常标题前有章节序号的内容称为论文的正文。正文之前和正文之后的内容，如目录、摘要和参考文献等，都是为正文服务的工具，它们的作用就是帮助读者更便利地了解和阅读正文。

10.1 摘要

论文的正文之前都要求有摘要，学位论文还要求有中文和英文两种摘要。摘要是对正文内容不加注释和评论的简短概述，它本身应是一篇完整的短文，可独立使用，比如被其他文件引用等，所以在摘要中要尽量避免出现脚注、交叉引用、数学式以及非通用的符号或术语等内容。

10.1.1 摘要环境

文类 `report` 和 `article` 都提供有编写摘要的 `abstract` 摘要环境：

```
\begin{abstract}
摘要内容
\end{abstract}
```

该环境可以自动生成用粗体字居中排版的摘要标题：Abstract，如果希望改换这个标题，例如改为 Summary，可对其标题名命令重新定义：`\renewcommand{\abstractname}{Summary}`。如果调用了中文标题宏包 `ctexcap`，该标题将自动改为黑体中文：摘要。若希望保持原有英文标题，可使用该宏包可选参数中的 `nocap` 选项，也可对摘要标题名命令重新定义。

文类 `report` 默认摘要环境排版的论文摘要为单独页，即进入摘要环境就另起一页，退出时再换一页；摘要标题页没有页码，如果摘要为多页，从第二页起页码从 2 开始计数，摘要之后内容的页码再从 1 开始计数。如果不需要摘要为单独页，可使用文类 `report` 可选参数中的 `notitlepage` 选项。

文类 `article` 默认摘要环境排版的论文摘要紧接上下文，不另起新页；如果希望摘要为单独页，可启用该文类的 `titlepage` 选项，这时摘要的页码与 `report` 默认时的相同。

文类 `book` 并没有提供摘要环境，这是因为学位论文等中长篇论著的摘要都是自成一章，通常采用章命令 `\chapter*` 来生成摘要标题并编写摘要内容。

摘要环境 `abstract` 或者带星号章命令 `\chapter*` 所生成的摘要标题，在默认情况下是不会被编入章节目录和进入页眉的。如果希望将摘要标题也写进目录和页眉，则需要分别使用 `\addcontentsline` 和 `\markboth` 命令。

10.1.2 自定义摘要环境

文类 `report` 和 `article` 所提供摘要环境的排版格式相同而且难以修改，如果对现有摘要环境 `abstract` 的排版格式不满意，可自定义一个排版格式能够灵活设置的摘要环境。

例 10.1 利用系统提供的通用列表环境 `list` 来自定义一个 `Abstract` 摘要环境，要求摘要标题为黑体、摘要内容为楷体、每段首行默认缩进为两个汉字的宽度并能调整、摘要文本左右缩进宽度相等且可设定。

第 11 章 编译

将源程序转变为目標程序的过程称为编译，这种转变需要相关的编译程序。对于 LaTeX 排版系统，它既是一种编程语言，同时又是这种语言的“翻译”，即它最终就是要将用 LaTeX 编程语言编写的源文件经过编译转变为 PDF 文件，这一过程也就是通常所说的排版。对源文件的编译过程主要可分为以下几个步骤。

(1) 词法分析，对由字符组成的单词进行处理，从左至右逐个字符地对源文件进行扫描，产生一个一个的单词符号。

(2) 语法分析，分析单词符号串是否形成符合语法规则的语法单位，如命令、环境和表达式等，检查它们之间的语法关系是否正确。

(3) 错误处理，在编译过程中如果发现源文件有错误，尽量纠正或限制其影响范围，尽可能避免中断编译。

(4) 目标代码，这是编译的最终目的，即生成能够被 SumatraPDF、Reader 或者 Acrobat 等 PDF 阅读器识别的 PDF 格式文件。

(5) 信息记录，将编译过程中的各种信息保存到辅助文件中。在每次编译后至少要输出两个辅助文件：一个是 .aux 引用记录文件，它记录了全文中所有交叉引用信息，以便为再次编译时提供相关数据；另一个是 .log 编译过程文件，它记录了整个编译过程，其中包括错误信息与警告信息，可作为源文件的修改依据。

11.1 编译方法

将源文件转变为 PDF 文件的编译方法有很多种，其中最常用的主要有三种，它们的编译过程如图 11.1 所示。

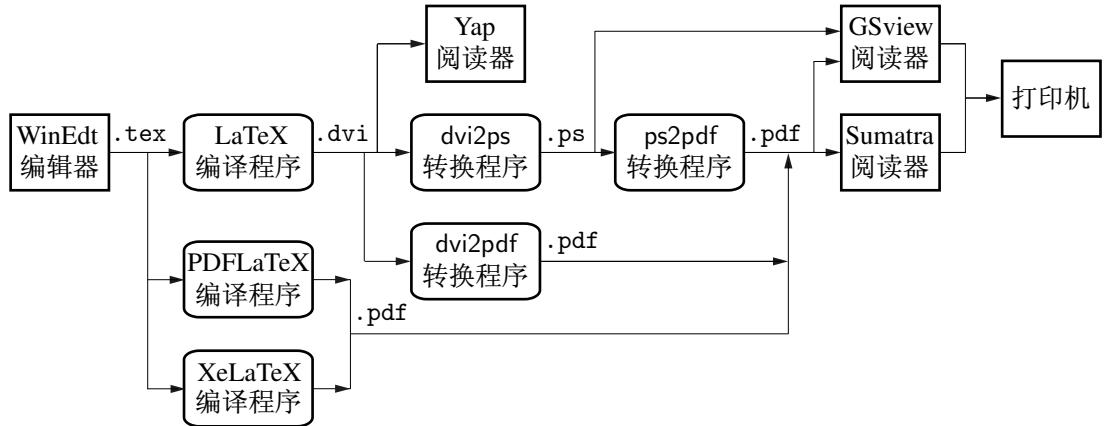


图 11.1 源文件编译过程示意图

从图 11.1 中可以看出，从最初的使用文本编辑器 WinEdt 编辑产生的 .tex 源文件至最终的 .pdf 文件分别有 LaTeX、PDFLaTeX 和 XeLaTeX 三种编译途径，其中经 LaTeX 编译后，还要再经过 dvi2ps 和 ps2pdf 转换程序，或者 dvi2pdf 转换程序才能生成 .pdf 文件，而另两种编译方法可以直接得到 .pdf 文件。各种编译方法其实就是对 LaTeX 源文件的不同解释，不同排版方式，其结果也各不相同。

第 13 章 幻灯片 — beamer

学位论文答辩或学术会议上通常都有 5~20 分钟的论文陈述，为了在这么短的时间内使听众能够迅速准确地了解论文的主要内容和研究成果，在陈述过程中放映幻灯片是非常直观、生动和高效的方法。

采用 LaTeX 制作幻灯片主要有两个好处，其一它是 PDF 格式，可不受操作系统和应用软件的约束，在任何计算机中都能放映；其二是它优异的数学式排版功能，可将论文中的各种复杂的数学式直接搬到幻灯片中。

现在已经有多种可以用于制作幻灯片的文类或者宏包，例如 beamer、foiltex、seminar、prosper、pdfslide、pdfscreen 和 slides 等，它们各有特色，其中较为常用的是 beamer 幻灯文类，它的特点主要如下。

(1) 幻灯片的源文件可使用 PDFLaTeX、XeLaTeX 或 LuaLaTeX 编译，直接生成 PDF 格式的幻灯文件，也可使用 $\text{LaTeX} \rightarrow \text{dvi2ps} \rightarrow \text{ps2pdf}$ 的编译方法，但不支持 $\text{LaTeX} \rightarrow \text{dvi2pdf}$ 的编译方法。任何计算机的应用软件中都有 PDF 格式文件的阅读器，所以走到哪里都不用担心幻灯片的放映问题。

(2) 绝大部分 LaTeX 命令仍然有效：章节目录用 `\tableofcontents` 生成，`itemize` 环境创建常规列表，使用 `\section` 和 `\subsection` 的层次结构。

(3) 提供大量多种类型的主题样式，可方便地更改幻灯片的整体风格，或是对某一局部的样式、字体和颜色等细节进行修改。

(4) 具有多种动画功能，使用和调整也很方便，可形象生动地演示各种过程的分解动作，有助于加深印象和理解。

本章专门介绍使用幻灯文类 beamer 制作陈述幻灯片。

13.1 基本结构

幻灯文类 beamer 是由 Till Tantau 编写的，是 CTeX 附带的一个用于制作幻灯片的文类，用它制作幻灯片与用 book 文类写论文一样，其源文件也是分为导言和正文两个部分，大部分 LaTeX 命令和环境都可以照搬到 beamer 中；由于幻灯片的特殊性，beamer 自身又提供了大量的专用命令和环境，以便于对幻灯片中的各种细节进行修饰。

使用文档类型命令 `\documentclass` 调用 `beamer.cls` 幻灯文类文件时，`amsfonts`、`amsmath`、`amssymb`、`amsthm`、`enumerate`、`geometry`、`graphics`、`graphicx`、`hyperref`、`ifpdf`、`keyval`、`xcolor`、`xxcolor` 和 `url` 等多个相关用途的宏包也同时被自动加载，所以在制作幻灯片时不必再单独调用这些宏包，以免发生冲突。

例 13.1 用一个最简单的例子来说明使用 beamer 幻灯文类制作中文陈述幻灯片的源文件基本结构。

```
\documentclass[14pt,hyperref={CJKbookmarks=true}]{beamer}
\usepackage[space,noindent]{ctex}
\usetheme{AnnArbor}
\setbeamercolor{normal text}{bg=black!10}
\begin{document}
```

後記

壬辰年九月

胡偉

本書在編寫時既想面面俱到又希望簡短精煉、不致連篇累牘使人望而却步。書中介紹的宏包都是在論文寫作時最常用的。但隨着老宏包的不斷改進、新宏包的不斷涌現、這些最常用的宏包已不見得是使用最方便、功能最強大的、其中一些宏包、儘管我們還在使用、已逐漸並不斷地被其他宏包所超越所替代、由於篇幅所限、這些後起之秀祇能另外著文介紹。書中對各種宏包的介紹都是選其主要功能、而有些宏包的輔助功能甚至連宏包本身的說明文件都未必說得很清楚。例如某個宏包命令祇是一筆帶過、甚至都沒提到、祇有當打開該宏包文件時才發現還有這麼一條很有用的命令。所以說爲了能够更準確地使用宏包、更充分地發揮宏包的效用、讀者還需要詳細閱讀宏包說明以及宏包文件。很多人抱怨命令太多很難記憶、急用時又找不到或找不全。爲此本書盡量將系統命令、環境命令、符號命令和宏包命令分門別類集中列表說明、比如長度命令、表格命令等。所以祇要通讀一遍本書、了解大致內容就可以了、在論文寫作時、需要哪方面的命令就去查閱相關的列表和說明、就如同查字典。如果是初學者、祇要看完前五章就可以實用了、其他各章可根據需要隨時查閱、它們之間沒有必然聯繫。書到用時方恨少、有些讀者可能會覺得意猶未盡、例如圖形繪制、宏包編寫等書中都未深入涉及、這方面的內容打算今後有機會另寫專文介紹。本書祇能陪您到此、以後就看您的了。

本篇後記是按照中華書局一九六二年出版的《古文觀止》一書的版式編寫，採用XeLaTeX編譯。

其源文件收錄于隨書附帶的資料光盤。