

L^AT_EX 科技排版

排版基础 + 模板使用

潘建瑜

华东师范大学 数学系

2017

讲座安排

上讲：排版基础 + 模板使用

下讲：数学公式 + 幻灯片

内容提要

- ① T_EX 与 L^AT_EX
- ② L^AT_EX 排版基础
- ③ 准备文档
- ④ 排版细节
- ⑤ 模板使用

T_EX 是什么

T_EX 是一种 **功能强大** 且 **十分灵活** 的排版语言

- T_EX 系统提供了 300 + 600 多条基本的排版命令
- T_EX 是目前公认的 **数学公式排版最好**的排版语言
- T_EX 是 **免费** 的
- T_EX 的名字来自大写的希腊字母 (τ, ϵ, χ), 意思是 “科技” 和 “艺术”

TeX 的缺点

- 命令繁多, 不便记忆
在使用中学习, 通过网络寻求帮助, 人性化的编辑器
- 错误难找: 慢慢积累经验
- 使用不直观
 - 要编译后才能显示效果
 - 每次修改后都要再次编译
 - 目前已有一些所见即所得的扩展, 如: LyX, TeXmacs



T_EX 的发展

- T_EX 的第一版于 1978 年面世

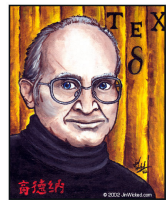
70 年代末, 斯坦福大学计算机系教授 D. E. Knuth (高德纳, 现代计算机科学先驱, 1974 年获图灵奖) 在看到其巨著 “The Art of Computer Programming” 第二卷的校样时, 对排版的低质量感到无法忍受, 决定开发一套高质量的排版系统, 于是就有了 T_EX



- 1982 年版的 T_EX 已是一个十分稳定的版本

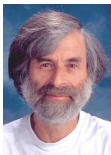
- 1990 年推出 3.1 版, 并宣布不再更新 (只修正 bug)

- 最新版本号: 3.14159265 (2014 年) → π



TeX 扩展 / 排版引擎 (部分)

- TeX 提供的命令都是一些很底层的命令, 普通用户使用起来不太方便
- TeX 扩展: 在 TeX 基础上, 定义新的命令, 为普通用户排版提供方便
 - Plain TeX : 由 Knuth 开发, 新定义 600 多条命令, 是通常所说的 TeX
 - $\mathcal{A}\mathcal{M}\mathcal{S}$ -TeX : 由美国数学会开发, 适合排版各种复杂的数学公式
 - L^ATeX (1984) : 由 Lamport 开发, 适合论文书籍, 对 TeX 推广贡献巨大
 - pdfTeX (1997): 由 H.T. Thanh 开发, 直接输出 PDF, 支持多种图片格式
 - xeTeX (2005): 由 J. Kew 开发, 支持 Unicode, 可直接访问操作系统字体
 - LuaTeX (2007): 由 T. Hoekwater 开发, 是 Lua 脚本语言和 TeX 的结合



L^AT_EX 介绍

- 1984 年前后, Lamport 开发了 L^AT_EX /'la:tek/ /'leitɛk/
- L^AT_EX 提供了一套针对文档逻辑结构的控制序列, 让 T_EX 变得更好用了, 普通用户即便不是很了解 T_EX, 也可以在很短的时间制作出高质量的排版结果, 因此迅速在计算机科学、数学及相关学科领域得到推广应用

L^AT_EX 2_ε



L^AT_EX 的特点

- 继承了 T_EX 的优点和缺点
- 自动编号: 章节、图表、公式定理、文献、...
- 自动生成目录、索引
- 公式、定理、参考文献、插图、页码等可以交叉引用
- 可以通过添加各种宏包扩展其功能, 实现各种特殊要求
-

L^AT_EX 和 Word

- Word 简单易用, 所见即所得 → 普通办公文档建议用 Word
- L^AT_EX 输出美观, 质量高 → 科技排版 (特别是数学) 推荐使用 L^AT_EX
- L^AT_EX 能实现 Word 的所有功能, 定制性高, 但易用性不如 Word

- T_EX 主要用户群: 高校和科研院所的科研人员和学生
- 主要学科: 数学, 力学, 物理, 计算机等
- 接受 T_EX 投稿的出版机构
 - AMS (美国数学会)
 - SIAM (美国工业与应用数学学会)
 - Springer
 - Elsevier
 - IEEE
 - Cambridge University Press
 - Kluwer Academic Publishers
 - Wiley
 - World Scientific
 -

中文 T_EX 排版

- CCT → 最早支持简体中文的 T_EX 扩展, 由中科院张林波教授开发, 比较符合中国人的习惯和时下中国印刷界的现行标准
- TY → 由华东师范大学肖刚、陈志杰等教授开发
- CJK → 由德国 L^AT_EX 开发, 可以同时处理中、日、韩三国文字
- ctex 宏集: 面向中文排版的通用 L^AT_EX 排版框架

英文排版建议: pdfL^AT_EX 或 xeL^AT_EX

中文排版建议: xeL^AT_EX + ctex 文档类

TeX 的发行版

- Windows 系统：TeXLive, MiKTeX, CTEX 套装
- Unix/Linux 系统：TeXLive
- Mac OS 系统：TeXLive, MacTeX

软件获取和安装

- Texlive：<http://tug.org/texlive/> (大而全，全部宏包)
- MiKTeX：<https://miktex.org/> (带部分宏包，其他宏包需在线安装)
- 编辑器：WinEdt, Texmaker, Texworks, Texstudio, vim, emacs, ...

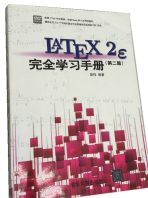
安装演示

- 以 Texlive 为例

学习资料

● 入门

- 📎 [The Not So Short Introduction to L^AT_EX 2_ε](#), 2015 (有中文版)
- 📎 [L^AT_EX 2_ε 完全学习手册](#), 2nd, 胡伟, 2013
- 📎 [L^AT_EX 入门](#), 刘海洋, 2013



● 提高

- [The T_EX Book](#), Knuth, 1993
- [L^AT_EX: A Document Preparation System](#), 2nd, Lamport, 1994
- [package references](#)

● 遇到问题怎么办？

- 丰富的网络资源
- 周围熟悉 L^AT_EX 的同学朋友



① T_EX 与 L^AT_EX

② L^AT_EX 排版基础

③ 准备文档

④ 排版细节

⑤ 模板使用


```

WinEdt 10.2 | MiKTeX (Unregistered Copy) - [C:\zzzzz\tex\beamer\sample_xelatex_article_utf8.tex]
File Edit Search Insert Document Project View Tools Macros Accessories TeX Options Help
FDLER.tex | Latex_Talk01.tex | sample_xelatex_article_utf8.tex | tik_xelatex.tex
%
% 建议用 pdflatex 或 xelatex 编译
%
\documentclass{article}

%==== 页面设置 =====
\usepackage[a4paper,top=2.54cm,bottom=2.54cm,left=3.17cm,right=3.17cm,%
includehead,includefoot]{geometry}

%==== 常用宏包 =====
\usepackage{amsmath,amssymb,amsfonts,amsthm}
\usepackage{graphicx}
\usepackage{subfigure}
\usepackage{float}
\usepackage{xcolor}
\usepackage[numbers,square,sort&compress]{natbib}
\usepackage{hyperref}
\hypersetup{colorlinks,citecolor=blue,linkcolor=blue}
\usepackage{booktabs}
\usepackage{caption}
\usepackage{epstopdf}

%==== 页眉页脚 =====
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\sectionmark}[1]{\markboth{\uppercase{#1}}{}}
\chead{\leftmark}
\cfoot{\thepage}

%==== 行间距 =====
\renewcommand{\baselinestretch}{1.1}

%==== 数学公式 =====
\allowdisplaybreaks

%==== 定义定理 =====
\newtheorem{theorem}{Theorem}[section]
\newtheorem{lemma}{Lemma}[section]
\newtheorem{corollary}[theorem]{Corollary}
\newtheorem{proposition}[theorem]{Proposition}
\newtheorem{definition}{Definition}[section]

```

Latex_Talk01.pdf | sample_xelatex_article_utf8.pdf
页码: 1 / 5

TITLE OF THE PAPER TITLE OF THE PAPER

First Author * Second Author †

Abstract

The main aim of this paper is to develop a fast algorithm for solving p... from image restoration. Abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract abstract.

1 Introduction

Linear systems with Toeplitz and Toeplitz-related coefficient matrices arise in many applications; see [10]. While many efficient algorithms have been developed for solving systems with Toeplitz structure, a few emerging applications lead to Toeplitz-related systems for which the available algorithms are not directly applicable [5, 10]. In this paper, we propose a preconditioned iterative method for weighted Toeplitz regularized least squares

$$\min_{x \in \mathbb{R}^n} \|Bx - b\|_2^2,$$

where the rectangular coefficient matrix B and the right-hand side b are of size $m \times n$ and m , respectively, with $m > n$.

$$B = \begin{bmatrix} \Xi K \\ \sqrt{\mu} I \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} \Xi f \\ 0 \end{bmatrix}.$$

Here $K \in \mathbb{R}^{m \times n}$ ($m \geq n$) is a full-rank Toeplitz matrix, $\Xi \in \mathbb{R}^{m \times m}$ is a symmetric positive definite matrix (as a weighting matrix), I is the identity matrix, $f \in \mathbb{R}^m$ is the right-hand side, and $\mu > 0$ is a regularization parameter [11].

Applications leading to such regularized Toeplitz least squares problems arise in image restoration with colored noise [7, 9] and nonlinear in the data. In these applications, the size of the problems can be very large, easily over 100,000. Because of the spatially variant property of weighted Toeplitz matrices ΞK , the rank of B can be very large. Efficient and effective preconditioners need to be designed.

L^AT_EX 源文件的基本框架

```
1 \documentclass{article} % 指定文档类型
2 %
3 % 导言区：全局设置，宏包调用等
4 %
5 \begin{document}
6 % 正文部分
7
8 Hi, this is my first \LaTeX{} file.
9
10 \end{document} % 结束
```

- L^AT_EX 源文件: 正文 + 命令 + 注解
- 反斜杠开头的字符串: 排版命令 (简称 命令)
- 注解符: 百分号 %
- `\documentclass{...}` 指定文档类型 (论文, 书籍, 幻灯片, 海报)
- `\begin{...}` 和 `\end{...}` 构成一个 环境

L^AT_EX (排版) 命令

```
\command
```

```
\command[option]{arguments}
```

- 方括号中的参数是可选的 (称为 选项), 花括号中的参数是必需的

例: 一些常用命令

- \documentclass, \title, \author, \date, \usepackage
- \begin{环境名}, \end{环境名} → 组成一个环境

- 定义新命令

```
\newcommand{新命令}{命令内容}
```

```
\renewcommand{已有命令}{命令内容}
```

```
1 \newcommand{\eps}{\varepsilon} % $\eps$ → ε
```

两个重要概念: 分组和环境

● 分组

- 有些命令只对其参数起作用, 如 `\textbf{abc}`
- 有些命令对后面所有的内容都起作用, 这些命令通常也称为声明, 如 `\bfseries`
- 可以利用大括号 (即分组) 来限制声明的作用范围

```

1 This is \textbf{bold face} style.\\
2 This is \bfseries bold face style.\\
3 This is {\bfseries bold face} style.

```

```

This is bold face style.
This is bold face style.
This is bold face style.

```

环境

● 环境

- 某些具有特定格式的内容需要放在相应的 **环境** 中, 如表格, 数学公式等

```
\begin{环境名}  
:  
\end{环境名}
```

- 环境中的排版命令, 通常只在该环境中起作用
- **document** 是 L^AT_EX 的一个最基本的环境
一篇文档有且只能有一个 **document** 环境
正文的所有内容都必须放在 **document** 环境中

选择文档类与导言区

```
\documentclass[选项]{文档类}
```

- 位于源文件的最前面, 用于指定文档的整体结构和布局, 必须且只能选一种
 - 常用 **文档类**: `article`, `book`, `beamer`, `ctexart`, `ctexbook`, `ctexbeamer`
 - 常用 **选项**:
 - 10pt(缺省值), 11pt, 12pt → 指定基本字体的大小
 - letterpaper(缺省值), a4paper, a5paper, ... → 指定纸张的大小
 - 单双面选项: `oneside`, `twoside`, `openright`, `openany`
 - 数学公式: `leqno` (公式编号在左边), `fleqn` (靠左显示行间公式)
- **导言区**: `\documentclass` 和 `\begin{document}` 之间的区域
 - 导言区用于放置 **全局控制命令**, 如: 调用宏包, 设置页面大小, ...
 - 放在导言区的命令对整个文档都起作用

L^AT_EX 举例

```
1 \documentclass[12pt,a4paper]{article}
2
3 \usepackage{amsmath} % AMS 数学公式 宏包
4 \usepackage{amssymb} % AMS 数学符号 宏包
5 \usepackage{amsfonts} % AMS 数学字体 宏包
6 \usepackage{graphicx} % 插图 宏包
7 \usepackage{xcolor} % 彩色 宏包
8
9 \begin{document}
10
11 The Euler equation is given by
12 $$ e^{ix} \triangleq \cos(x) + i\sin(x). $$
13
14 \end{document}
```



换行, 分段, 分页

- **换行**: 自然换行 (若需 **强制换行**, 可使用 `\` 或 `\linebreak`)
 - ☞ 一般情况下, 不建议使用强制换行
- **分段**: 一个空行或 `\par`
 - ☞ 建议使用空行进行分段
- **分页**: 自然分页, 若需 **强制分页**, 可用 `\newpage`, `\clearpage` 或 `\pagebreak`
 - ☞ 一般情况下, 不建议使用强制分页.
- **行间距**: 行间距伸展因子 `\baselinestretch` 或伸展命令 `\linespread`

```
\renewcommand{\baselinestretch}{1.2}
\linespread{1.2}
```
- **段落间距和段落缩进**: 用自动设定的即可
 - ☞ 英文每节的第一段首行不会自动缩进.

长度

mm 毫米	in 英寸	em 大约为大写字母 M 的宽度
cm 厘米	pt 点 / 磅	ex 大约为小写字母 x 的高度

 1 in = 2.54cm = 72pt, em 和 ex 与当前字体尺寸有关

- 长度设置与修改

```
\setlength{长度数据命令}{长度}
\addtolength{长度数据命令}{长度}
```

- 常用长度数据命令 (变量)

- \textwidth, \textheight
- \parindent, \parskip, \baselineskip

```
1 \setlength{\textwidth}{15cm}
2 \setlength{\textheight}{21cm}
3 \setlength{\hoffset}{-5mm} % 长度有时也可以是负值
```

水平间距

- 强制空格: `\u`
- `\quad` → 产生一段宽度为 1em 的水平空白
- `\qqquad` → `\quad` 的两倍
- `\,` → 大约为 `\quad` 的 3/18
- `\hspace{宽度}` → 产生指定宽度的水平空白
- `\hspace*{宽度}` → 若要在行首产生一定的空白, 则需使用此命令
- `\hfill` → 根据排版需要插入空白, 撑满整行
- `\hphantom{文本内容}`: 水平空白的宽度等于文本内容的总宽度

垂直间距

- `\smallskip` → 垂直空白高度为 3pt plus 1pt minus 1pt
- `\medskip` → `\smallskip` 的两倍
- `\bigskip` → `\smallskip` 的四倍
- `\vspace{高度}` → 产生指定高度的垂直空白
- `\vspace*{高度}` → 同 `\vspace`, 主要同在页面的顶部
- `\vphantom{文本内容}` → 垂直空白的高度等于文本内容的总高度

特殊字符

- 有 10 个字符被赋予了特殊用途, 需要使用相应的命令才能输出

字符	#	\$	%	{	}	~	_	^	&	\
命令	<code>\#</code>	<code>\\$</code>	<code>\%</code>	<code>\{</code>	<code>\}</code>	<code>\~{}</code>	<code>_{} </code>	<code>\^{} </code>	<code>\&</code>	<code>\textbackslash</code>

- 符号 “>”, “<”, “|” 被定义成数学符号, 只能用在数学模式中, 若要在普通文本中输出, 需使用相应的命令

字符		<	>
命令	<code>\textbar</code>	<code>\textless</code>	<code>\textgreater</code>

- 引号与连字号

字符	‘	’	“	”	`	-	-	—
命令	<code>`</code> (倒引号)	<code>'</code>	<code>``</code>	<code>"</code> 或 <code>'</code>	<code>\`{}</code>	<code>-</code>	<code>--</code>	<code>---</code>

一些注意事项和建议

- 中文和英文 之间, 中文与数学公式 之间 → 用空格隔开
- 各种环境的开始和结束 → 独占一行
- 自然分段 → 建议使用一个空行
- 自然换行 → 慎用强制换行 “`\\`”
- 标点符号一致 → 全部采用中文标点, 或者全部采用英文标点
- 数学公式中的标点 → 必须用英文标点
- 单个回车 → 编译时作为一个空格处理
- 多个连续的空格 → 编译时作为一个空格处理
- 编译出错时如何终止编译 → `x`
- 如何清空辅助文件

① T_EX 与 L^AT_EX

② L^AT_EX 排版基础

③ 准备文档

④ 排版细节

⑤ 模板使用

准备文档

- 文档 (论文, 书籍) 基本框架
 - 标题
 - 摘要
 - 目录
 - 正文: 章, 节, 小节, . . .
 - 附录, 参考文献, 索引
- 文档 (论文, 书籍) 基本元素
 - 普通文本: 字体, 大小, 脚注
 - 列表, 表格, 插图
 - 颜色, 链接
 - 数学公式

文档类缺省设置

```
\documentclass[选项]{文档类}
```

- **article** 文档类缺省设置
 - 标题, 摘要, 正文 (篇, 节, 小节, ...), 参考文献; 不区分奇偶页, 没有章
- **book** 文档类缺省设置
 - 标题页 (书名等, 独占一页), 目录, 正文 (篇, 章, 节, ...), 参考文献, 索引
 - 区分奇偶页, 奇数页也称 **右页**, 偶数页也称 **左页**, 采用不同的版式
- 页面布局 (页面宽度, 页面高度, 页眉高度, 页脚高度, 各种边距等等)
 - 建议使用 **geometry** 宏包 (详细用法参见宏包手册)
- 页眉页脚

```
\pagestyle{版式} % 四种版式: empty, plain, heading, myheading
\thispagestyle{版式} % 仅改变当前页的版式
```

🔧 高级定制: **francyhdr** 宏包 (详细用法参见宏包手册)

章节命令

```
\part  
\chapter,  
\section, \subsection, \subsubsection,  
\paragraph, \subparagraph
```

- `article` 类中没有 `\chapter`
- 章节命令的一般调用方法

```
\章节命令[短标题]{标题}  
\章节命令*{标题}
```

- 短标题: 用于显示在目录和页眉中, 缺省与标题相同
- 带星号的章节命令: 不参与自动编号

章节自动编号

- 每类章节都被赋予一个层次号: 层次号越小, 级别越高
 - `\chapter` 的层次号为 0, `\section` 为 1, 向后逐次递增
(`\subsection` 为 2, `\subsubsection` 为 3, ...)
 - `\part` 在 book 中为 -1, 在 article 中为 0

- book 类中的自动编号的章节有:
 - `\part`, `\chapter` → 独立编号
 - `\section`, `\subsection` → 关联编号, 如 2.1, 2.2.1
(节的编号前带有章的编号, 小节的编号前面带有章和节的编号)

- article 类中的自动编号的章节有
 - `\part`, `\section` → 独立编号
 - `\subsection`, `\subsubsection` → 关联编号

- 可根据需要修改自动编号所到达的最深层次数

```
\setcounter{secnumdepth}{数}
```

- book 类: 取值范围 -2 至 5, 缺省为 2, 当值为 -2 时取消所有编号
- article 类: 取值范围 -1 至 5, 缺省为 3, 当值为 -1 时取消所有编号

```
\setcounter{secnumdepth}{4}
```

→ subsection 和 paragraph 都自动编号

- 自动编号的章节都有各自所对应的计数器

- 计数器从属关系: part → chapter → section → subsection → ...
- 可以使用 \setcounter 来修改计数器的值

```
\setcounter{计数器名}{数}
```

- 每次章节命令时, 计数器自动加 1, 并同时 will 将低级别的计数器置零

章节举例

Part I

篇 (Part) 的标题

1 节 (Section) 的标题

1.1 小节 (Subsection) 的标题

1.1.1 子节 (Subsubsection) 的标题

article

Chapter 1

章 (chapter) 的标题

1.1 节 (Section) 的标题

1.1.1 小节 (Subsection) 的标题

子节 (Subsubsection) 的标题

book

标题部分: 标题, 作者, 日期, 脚注等

```
\title{标题}
\author{作者}
\date{日期}
\maketitle
```

- 以上命令只提供生成标题所需的信息, 并不生成标题!
- 生成标题的命令: `\maketitle` → 这个不可省略
- `\date{}` → 不输出日期
如果省略 `\date` 命令, 则自动加上当前日期
- 标题部分的角注: `\thanks{...}`
必需放置在 `\title`, `\author` 和 `\date` 的 **参数** 中 (即 `{}` 内), 可出现多次
- 标题过长可以用 `\\` 强制换行
- 若有多个作者, 可用 `\and` 隔开 (横排), 也可使用 `\\` (竖排)

摘要

```
\begin{abstract}  
... ..  
\end{abstract}
```

- book 文档类没有 摘要

生成目录

```
\tableofcontents
```

- 设置目录中显示的章节层次: `\setcounter{tocdepth}{层次号}`
- 只有自动编号的章节才会在目录中显示 (带星号的章节不显示)
- ☞ 若需在目录中输出不自动编号的章节, 可使用下面的命令

```
\addcontentsline{toc}{章节名称}{条目内容}
\addtocontents{toc}{条目内容}
```

章节名称 可以是 `part`, `chapter`, `section`, ...

第二个命令按原样输出条目内容, 且不显示页码

- 为了生成正确的目录, 有时需编译两次
- 章节标题和目录的高级定制: `titlesec` 和 `titletoc` 宏包

参考文献环境: thebibliography

```
\begin{thebibliography}{编号样本}
\bibitem[编号]{标签} 文献条目
\bibitem[编号]{标签} 文献条目
  ⋮
\end{thebibliography}
```

- “编号” → 通常省略，系统自动按顺序编号，如 [1]，[2]，...
- “编号样本” → 指定用多大地方显示“编号”，一般为数字，位数等于最大编号的位数
- 标签 → 文献的 id，可以由字母，数字和除逗号外的符号组成
 - 每个文献的 标签 必须唯一 (互不相同)
 - 文献的引用: `\cite{标签}`, `\cite{标签1, 标签2}`
- 文献条目: 论文 (作者, 标题, 期刊, 卷期, 年代, 页码),
书籍 (作者, 书名, 出版社, 年代)
- 参考文献的高级定制: `natbib` 宏包

```
1 \begin{thebibliography}{99}
2
3 \bibitem{AH77}
4 \newblock H. Andrews and B. Hunt,
5 \newblock \emph{Digital Image Restoration},
6 \newblock Prentice-Hall, Englewood Cliffs, NJ, 1977.
7
8 \bibitem{KS95}
9 \newblock T. Kailath and A. H. Sayed,
10 \newblock Displacement Structure: Theory And Applications,
11 \newblock \emph{SIAM Rev.}, 37 (1995), 297--386.
```

交叉引用

- 交叉引用: 章节, 图表, 定理, 引理, 公式, 页码, 参考文献, ...
- L^AT_EX 自动跟踪所有的变动, 保证所有交叉引用与索引记录的正确性
- 相关命令
 - 设置标签: `\label{标签}`
 - 章节, 图表, 定理等的引用: `\ref{标签}`
 - 公式的引用: `\eqref{标签}`
 - 页码的引用: `\pageref{标签}`
 - 参考文献的引用: `\cite{标签}`

索引

● 索引: `makeidx` 宏包

```
\usepackage{makeidx}
```

```
\makeindex → 激活索引功能, 该命令不可缺省
```

- 创建索引条目: 在需要索引的地方插入命令 `\index{索引条目}`
- 排印索引 → `\printindex`, 通常出现在文档最后
- 编译过程: 先用 L^AT_EX 编译, 再用 `makeindex` 编译
- 索引按 ASCII 码排序

● 多索引问题: `index` 宏包, 详细用法参见宏包手册

① T_EX 与 L^AT_EX

② L^AT_EX 排版基础

③ 准备文档

④ 排版细节

⑤ 模板使用

排版模式: 文字模式与数学模式

● 文字模式

- 段落模式: 自动分行, 分段, 分页
- 左到右模式: 将输入字符排成一排, 不换行, 无论长短

● 数学模式: 排版数学公式

- 数学公式一定要放在数学环境中

● 文本对齐方式 (缺省为左对齐)

- 左对齐: `\raggedright` 或使用 `flushleft` 环境
- 右对齐: `\raggedleft` 或使用 `flushright` 环境
- 文本居中: `\centering` 或使用 `center` 环境

字体命令

- 前面关于字体的命令都是 **声明**
 - 这些命令在遇到新的同一类命令之前一直起作用
 - 若想限制声明的作用范围, 可使用**分组** 或**环境**
- 每一个字体**声明**都有对应的带参数的**字体命令**
- 字体命令: 只对其参数中的文本起作用, 如:

`{\bfseries 这是黑体 }` \iff `\textbf{这是黑体}`

- 族: `\textrm{...}`, `\textsf{...}`, `\texttt{...}`
 - 形状: `\textup{...}`, `\textit{...}`, `\textsl{...}`, `\textsc{...}`
 - 系列: `\textmd{...}`, `\textbf{...}`
 - 采用缺省值: `\textnormal{...}`, 对应的声明为: `\normalfont`
 - 强调命令: `\emph{...}`, 对应的声明为: `\em`
- 缺省的西文字体: **中等权重, 直立的罗马字体**

字体的尺寸

● 字体的大小

- 基本尺寸: 10pt, 11pt 和 12pt, 缺省为 10pt

```
\documentclass[11pt]{article}
```

- 设定了字体的基本尺寸后, 可使用下面的声明来改变字体大小

<code>\tiny</code>	5pt	<code>\large</code>	12pt
<code>\scriptsize</code>	7pt	<code>\Large</code>	14.4pt
<code>\footnotesize</code>	8pt	<code>\LARGE</code>	17.28pt
<code>\small</code>	9pt	<code>\huge</code>	20.74pt
<code>\normalsize</code>	10pt	<code>\Huge</code>	24.88pt

- ☞ 以上尺寸是以 10pt 为基本尺寸时的大小

中文字体


● 中文字体 (ctex 文档类)

```
\heiti, \songti, \fangsong, \kaishu, \lishu, \youyuan, \yahei
```

● 字号

- 所有的西文字号命令对中文同样起作用, 如 `\huge`, `\LARGE`, `\Large`, ...
- ctex 文档类提供的字号命令: `\zihao{字号}`, 如 `\zihao{3}`, `\zihao{-4}`

0	初号 = 42 磅	3	三号 = 16 磅	6	六号 = 7.5 磅
-0	小初 = 36 磅	-3	小三 = 15 磅	-6	小六 = 6.5 磅
1	一号 = 26 磅	4	四号 = 14 磅	7	七号 = 5.5 磅
-1	小一 = 24 磅	-4	小四 = 12 磅	8	八号 = 5 磅
2	二号 = 22 磅	5	五号 = 10.5 磅		
-2	小二 = 18 磅	-5	小五 = 9 磅		

 1 英寸 = 72 磅 (pt) = 2.54cm

- 详情参见 `ctex` 宏包说明.

使用系统中的中文字体

- 使用系统中的中文字体

- 打开一个 `cmd` 窗口, 输入以下命令

```
fc-list :lang=zh-cn > zhfonts.txt
```

→ 打开文件 `zhfonts.txt` 即可看到系统中的中文字体

- 在 `tex` 文件中设置字体 (以 `STZhongsong` 华文中宋为例)

```
\setCJKfamilyfont{STSong}{STZhongsong} → 设置字体
```

```
\newcommand{\STSong}{\CJKfamily{STSong}} → 设置字体命令
```

`\STSong` 就可以作为字体命令使用了, 如 `{\STSong 这是华文中宋}`

表格环境: tabular

```
\begin{tabular}[竖向位置]{列格式}
  first line \\
  :
  last line \\
\end{tabular}
```

- **竖向位置**: 表格在竖直方向与外部文本行的相对位置, 取值为: `t` 或 `b`, 分别表示**上对齐**和**下对齐**, 缺省为**居中对齐**
- **列格式**: 用于指定各列的格式, 常用的参数有: `l`, `c`, `r`, `|`, `||`, ...
- 行与行之间用 `\\` 分隔, 每一行的列与列之间用 `&` 分隔
- 行与行之间的分界线:
 - `\hline`: 与表格同宽的水平线
 - `\cline{m-n}`: 从第 `m` 列开始到第 `n` 列结束的水平线
- 高级表格: `longtable`, `slashbox`, `colortbl`, `booktabs` 等宏包

列表环境

`itemize` → 带相同的标签

`enumerate` → 条目标签为自动编号的符号

`description` → 人工指定各条目的标签

● `itemize` 列表环境

```
\begin{itemize}
  \item[标签] 条目内容
  ... ..
\end{itemize}
```

- 缺省的标签与层数有关, 分别为: ■, -, *, ·
- 也可通过选项**标签**来指定标签
- 不要标签: `\item[]`

● enumerate 列表环境

```
\begin{enumerate}
  \item[标签] 条目内容
  ...
\end{enumerate}
```

○ 缺省标签为自动编号的符号, 与层数有关, 分别为:

- 第一层: 阿拉伯数字后跟圆点: 1. 2.
- 第二层: 圆括号包围的小写拉丁字母: (a) (b)
- 第三层: 小写罗马数字后跟圆点: i. ii.
- 第四层: 大写拉丁字母后跟圆点: A. B.

● 高级列表功能: list 环境, enumitem 宏包

脚注

`\footnote{脚注文本}` → 自动编号

`\footnote[标记]{脚注文本}` → 手工指定脚注标记

- 该命令应紧接在需要注释的文字后面, 排版后会在所在处显示一个脚注标记, 同时将脚注内容显示在当前页的底部, 并带有相同的脚注标记
- 脚注标记通常为上标形式的数字, 并自动编号

计数器

所有自动编号的条目都有相应的计数器, 如

part, chapter, ..., footnote, page, equation, ...

- 计数器的关联 (确定从属关系, 该命令需放在导言区)

```
\numberwithin{从计数器}{主计数器}
```

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

- 高级功能
定义新的计数器, 获取计数器的值, 设置显示方式, 修改计数器等, 参见相关资料

抄录环境

```
\begin{verbatim}  
... ..  
\end{verbatim}
```

```
\begin{verbatim*}  
... ..  
\end{verbatim*}
```

- 抄录环境: 按原样输出文本, 包括所有 L^AT_EX 命令
 - 带星号 的环境用 `\` 显示空格
- 👉 功能更丰富的抄录: `fancyvrb` 宏包, 提供 `Verbatim` 环境

算法与代码

`\usepackage{algorithm}` → 算法
`\usepackage{algpseudocode}` → 伪代码
`\usepackage{listings}` → 各种语言的代码

```
1 \begin{algorithm}
2 \caption{LU Decomposition}
3 \begin{algorithmic}[1]
4   \For{$k=1$ to $n-1$}
5     \For{$i=k+1$ to $n$}
6       \State  $a_{ik} = a_{ik}/a_{kk}$ 
7       \For{$j=k+1$ to $n$}
8         \State  $a_{ij} = a_{ij} - a_{ik}a_{kj}$ 
9       \EndFor
10    \EndFor
11  \EndFor
12 \end{algorithmic}
13 \end{algorithm}
```

颜色: xcolor 宏包

```
\usepackage{xcolor}
```

- 使用彩色

```
\color{颜色名} → 声明, 对后面的所有文本起作用
```













```
\textcolor{颜色名}{对象} → 命令
```

- 页面背景色: `\pagecolor{颜色名}`

- 彩色盒子: `\colorbox{颜色名}{对象}`

- 带框的彩色盒子: `\fcolorbox{边框颜色}{背景颜色}{对象}`

● 可以直接使用的颜色

 <i>black</i>	 <i>darkgray</i>	 <i>lime</i>	 <i>pink</i>	 <i>violet</i>
 <i>blue</i>	 <i>gray</i>	 <i>magenta</i>	 <i>purple</i>	 <i>white</i>
 <i>brown</i>	 <i>green</i>	 <i>olive</i>	 <i>red</i>	 <i>yellow</i>
 <i>cyan</i>	 <i>lightgray</i>	 <i>orange</i>	 <i>teal</i>	

● 更多颜色 (需加 `dvipsnames` 选项)


 <i>Apricot</i>	 <i>Cyan</i>	 <i>Mahogany</i>	 <i>ProcessBlue</i>	 <i>SpringGreen</i>
 <i>Aquamarine</i>	 <i>Dandelion</i>	 <i>Maroon</i>	 <i>Purple</i>	 <i>Tan</i>
 <i>Bittersweet</i>	 <i>DarkOrchid</i>	 <i>Melon</i>	 <i>RawSienna</i>	 <i>TealBlue</i>
 <i>Black</i>	 <i>Emerald</i>	 <i>MidnightBlue</i>	 <i>Red</i>	 <i>Thistle</i>
 <i>Blue</i>	 <i>ForestGreen</i>	 <i>Mulberry</i>	 <i>RedOrange</i>	 <i>Turquoise</i>
 <i>BlueGreen</i>	 <i>Fuchsia</i>	 <i>NavyBlue</i>	 <i>RedViolet</i>	 <i>Violet</i>
 <i>BlueViolet</i>	 <i>Goldenrod</i>	 <i>OliveGreen</i>	 <i>Rhodamine</i>	 <i>VioletRed</i>
 <i>BrickRed</i>	 <i>Gray</i>	 <i>Orange</i>	 <i>RoyalBlue</i>	 <i>White</i>
 <i>Brown</i>	 <i>Green</i>	 <i>OrangeRed</i>	 <i>RoyalPurple</i>	 <i>WildStrawberry</i>
 <i>BurntOrange</i>	 <i>GreenYellow</i>	 <i>Orchid</i>	 <i>RubineRed</i>	 <i>Yellow</i>
 <i>CadetBlue</i>	 <i>JungleGreen</i>	 <i>Peach</i>	 <i>Salmon</i>	 <i>YellowGreen</i>
 <i>CarnationPink</i>	 <i>Lavender</i>	 <i>Periwinkle</i>	 <i>SeaGreen</i>	 <i>YellowOrange</i>
 <i>Cerulean</i>	 <i>LimeGreen</i>	 <i>PineGreen</i>	 <i>Sepia</i>	
 <i>CornflowerBlue</i>	 <i>Magenta</i>	 <i>Plum</i>	 <i>SkyBlue</i>	

● 更多用法参见宏包指南

超链接: `hyperref` 宏包

```
\usepackage[选项列表]{hyperref}
```

- 在有交叉引用的地方 (如目录, 书签, 参考文献, 公式等) 建立链接
- 提供对外部文件, 互联网网址, 邮件地址的链接
- 常用选项 (也可通过 `\hypersetup{选项列表}` 来设置)
 - `bookmarks` → 创建书签, 缺省为 `true`
 - `CJKbookmarks` → 支持中日韩文字的书签
 - `colorlinks` → 使用彩色显示链接, 缺省为红色方框
 - `linkcolor` → 内部普通链接 (如页码) 的颜色, 缺省为 `red`
 - `citecolor` → 文献引用链接的颜色, 缺省为 `green`
 - `urlcolor` → URL 链接的颜色, 缺省为 `magenta`
 - `breaklinks` → 允许在链接中断行, 缺省不允许

 更多选项参见宏包手册

举例

```

1 \usepackage{hyperref}
2 \hypersetup{CJKbookmarks=true, % 支持中文书签
3     colorlinks=true, % 使用彩色链接
4     citecolor=blue, % 引用标记颜色
5     linkcolor=blue, % 内部普通链接的颜色
6     urlcolor=blue, % url 链接的颜色
7     breaklinks=true } % 允许在链接处换行

```

- 创建网页链接: `\url` 和 `\href`

```

\url{网址}
\href{网址}{文本}

```

- `\url` → 生成 [网址](#) 的同时在页面上输出其内容
- `\href` → 生成 [网址](#) 的同时在页面上输出 `文本` 的内容

graphicx 宏包

```
\usepackage{graphicx}
```

- 支持插入黑白或彩色图片
- 可以对图片进行裁剪, 缩放和旋转
- 支持的图片格式:
 - eps 格式: L^AT_EX 只支持 eps 图片
 - pdf, jpg, png 格式: 这是 pdfL^AT_EX 和 xeL^AT_EX 支持的格式
 - ☞ pdfL^AT_EX 和 xeL^AT_EX 不支持 eps 图片
 - ☞ epstopdf 宏包: 自动将 eps 图片转换为 pdf 图片
- 推荐使用 pdfL^AT_EX 和 xeL^AT_EX

插图基本命令

```
\includegraphics[选项]{图形文件名}
```

- 图形文件名 中可以含路径
- 常用选项有
 - width, height → 指定图形的宽度和高度
 - 👉 若只指定宽度或高度, 则按比例缩放图形
 - scale → 缩放因子, 如 scale=0.8
 - angle → 指定旋转角度, 逆时针, 以度为单位

插图举例

```

1 \includegraphics[scale=0.2]{tiger.png}
2 \includegraphics[width=2.5cm]{tiger.png}
3 \includegraphics[width=0.4\textwidth]{tiger.png}

```



浮动图表环境

```
\begin{figure}[位置]
```

```
... ..
```

```
\end{figure}
```

```
\begin{table}[位置]
```

```
... ..
```

```
\end{table}
```

- 浮动图表: 自动调整图表位置, 避免出现大片的空白
- 位置选项的取值: `h` → here, `t` → top, `b` → bottom, `p` → page
- 优先顺序: `h` → `t` → `b` → `p`
- 缺省值为 `tbp`
- 固定在当前位置: `H` → 需加载 `float` 宏包

自动编号与图表目录

- 添加标题, 并自动编号 (只有 **浮动图表** 才能自动编号)

```
\caption[短标题]{标题}
```

- 插图目录和表格目录

```
\listoffigures → 打印插图目录
```

```
\listoftables → 打印表格目录
```

- 通常放在 `\tableofcontents` 后面
- 高级功能: `float`, `caption`, `subfigure` 宏包


```
1 \documentclass[a4paper]{ctexbook}
2 \input{format.tex}
3
4 \begin{document}
5 \include{cover.tex}
6
7 \frontmatter
8 \tableofcontents
9 \include{preface.tex}
10
11 \mainmatter
12 \include{chap01.tex}
13 \include{chap02.tex}
14 ... ..
15
16 \backmatter
17 \include{reference.tex}
18
19 \end{document}
```


模板使用

- 论文 (英文, 中文)
- 书籍 (中文)
- 毕业论文 (中文)
- 幻灯片 (英文, 中文)
- poster