

CTEX FAQ (常见问题集)

Version 0.4 beta (86)

吴凌云 (aloft@ctex.org)

2004年9月27日 Rev. 1.22

本文的大部分内容都来自于 [CTEX 论坛](#) 上的讨论。版本号为 0.4，一共收录有 86 个问题。这份常见问题集会不断更新，请到 [CTEX 网站](#) 的主页下载它的最新版本。

本文档的主要目的是帮助你快速的找到问题的答案，因此可能不是非常详细。如果你想系统的学习 T_EX 和 L_AT_EX 请参考专门的资料（见 [问题 83](#) 和 [问题 84](#)）。



符号表示后面的内容需要比较深的 L_AT_EX 知识。



表示后面的内容带有试验性质，有时会带来意想不到的结果，应尽量避免在实际中使用。

新增 17 18 80 82

修订 3 20 38 39 43 52

Copyright ©2003–2004 C_TE_X and all the Contributors to C_TE_X FAQ. All rights reserved.

目录

A 关于 FAQ	5
1 C _T E _X 是干什么的？	5
2 C _T E _X 网站包括几个部分？	5
3 这个 FAQ 的目的是什么？	5
4 我可以打印、传播、转载这份文档的内容吗？	5
B 基本概念	5
5 什么是 T _E X？	5
6 L _A T _E X 是什么？	6
7 什么是 $\mathcal{A}\mathcal{M}\mathcal{S}$ -T _E X/ $\mathcal{A}\mathcal{M}\mathcal{S}$ -L _A T _E X？	6
8 T _E X 和 L _A T _E X 是什么关系？	6
9 L _A T _E X 2.09 和 L _A T _E X 2 _ε 有什么区别？	6
10 MiK _T E _X 、fp _T E _X 、te _T E _X 、C _T E _X 又是什么关系？	7
11 什么是 CTAN？	7

C 安装升级	7
12 哪里可以下载 C _T E _X 中文套装软件?	7
13 哪里可以下载 MiK _T E _X 的最新版本?	7
14 哪里能找到我想要的宏包?	8
15 如何升级现有的 MiK _T E _X ?	8
16 我下载了一些新的宏包, 该如何安装到我的系统中去?	8
17 C _T E _X 中文套装软件的 Basic、Pro、Full 版本都有什么区别?	8
18 C _T E _X -Fonts、C _T E _X -CS、C _T E _X -Ext 都是干什么的? 和 C _T E _X 中文套装有什么关系?	9
D 新手入门	9
19 我是新手, 该如何开始?	9
20 能否举一个简单的英文排版例子?	9
21 用什么编辑工具来编辑 L ^A T _E X 文件?	10
22 什么是导言区 (preamble)?	10
23 什么是文件名数据库? 怎么刷新?	10
E 中文支持	11
24 比较一下 CJK 与 CCT 的优缺点?	11
25 能否举一个简单的 CCT 中文排版例子?	11
26 能否举一个简单的 CJK 中文排版例子?	12
27 在 CJK 中如何使段首缩进两个中文字符宽度?	12
28 在 CJK 中如何调整汉字的间距?	12
29 CJK 和 CJK* 有何不同?	12
30 中英文混排时为何无法正确分行?	13
31 什么是点阵字库和矢量字库?	13
32 在 CJK 中都可以使用哪些矢量字库?	14
33 两种类型矢量字库会不会发生冲突?	14
34 如何生成支持中文拷贝粘贴的 PDF 文件?	14
F 文档排版	14
35 如何使首段段首也缩进?	14
36 如何设置双倍或者更大的行间距?	14
37 如何注释掉一大段内容, 不做编译?	14
38 如何生成分栏的文档?	15
39 如何使得双栏文档保持底部对齐?	15
40 如何在双栏文档中使用通栏的摘要?	16
41 如何产生全部横排的文档?	17
42 如何使文档中的某一页横排?	17
43 如何使用 New Time Roman 字体?	17

G 数学公式	17
G.1 编号	17
44 如何实现定理、定义、推论等的统一编号?	17
45 如何让公式编号中使用当前章节编号做前缀?	18
46 如何实现如 (1a) (1b) 这种公式编号?	18
47 如何让一个公式编号表示为另一个公式编号的变化形式?	19
G.2 字体	19
48 实数域 \mathbb{R} 或复数域 \mathbb{C} 等的字体该用什么命令?	19
49 书中向量常用斜体加黑体表示, 该用什么字体命令排版?	19
G.3 其它	19
50 如何使上下限出现在求和、积分符号的上下方而不是右边?	19
51 如何实现上下方能输入公式文字且可伸长的等号或者箭头符号?	20
H 图形表格	21
H.1 图形	21
52 L ^A T _E X 如何使用图形?	21
53 L ^A T _E X 支持哪些图形格式?	22
54 DVI 里不能显示 EPS 图片?	22
H.2 表格	22
55 表格中使用 p 参数的列如何改变对齐方式?	22
56 如何生成固定宽度的表格?	23
H.3 浮动对象	24
57 浮动对象和上下文之间的距离太大了, 怎么办?	24
58 分栏文档中如何使用单栏或者通栏浮动对象?	25
59 分栏文档中为何不能设定通栏浮动对象的位置?	25
I 参考文献	25
60 如何将文献引用作为上标出现?	25
61 如何排序并压缩连续的引用, 产生如 [2,4-8] 形式的引用?	26
62 如何定制自己的参考文献样式?	26
63 如何才能把参考文献条目放在正文中?	26
64 能否生成反向引用(从参考文献指向引用的正文)?	26
65 如何让每章或者每节都有单独的参考文献?	27
66 如何产生多个参考文献列表?	27
J 其它专题	28

J.1 脚注	28
67 如何在表格中使用脚注?	28
68 如何在章节标题中使用脚注?	29
69 如何在图表的标题中使用脚注?	29
70 如何重复使用相同内容的脚注?	29
J.2 目录	30
71 怎样象第几章一样, 将参考文献也加到目录?	30
72 手工加入的目录项页码错误, 生成的书签链接也不对, 怎么回事?	30
J.3 特殊符号	31
73 如何输出 $\hat{\ } $ 和 $\tilde{\ } $ 符号?	31
74 如何显示可见的空格符号 $_ $?	31
75 如何显示带圈的符号?	31
K 高级技术	32
76 <code>\makeatletter</code> 和 <code>\makeatother</code> 是干什么用的?	32
77 如何查询 L ^A T _E X 命令和参数的缺省定义?	32
L 辅助工具	32
78 WinEdt 和 GSview 经常出现过期和注册提示, 如何解决?	32
79 如何去掉 WinEdt 的自动断行和自动续行?	32
80 如何让 WinEdt 显示行号?	33
81 Yap 里如何预览带有颜色的 EPS 图片?	33
M 错误处理	33
82 Error: pdflatex (file simsun.ttf): can't find table 'maxp'	33
N 帮助文档	33
83 能否推荐一些比较好的中文资料?	33
84 能否推荐一些比较好的英文资料?	33
85 有没有关于 L ^A T _E X 宏包的详细说明?	34
86 我的问题没找到答案, 或者有答案但是解决不了我的问题, 怎么办?	34
参考文献	34
索引	36
Contributors	40
ChangeList of v0.4	41

A 关于 FAQ

1 C_TE_X 是干什么的？

C_TE_X 是一个关于 T_EX 的中文网站，C_TE_X 是中文 T_EX 的缩写。C_TE_X 的宗旨是为中国广大的 T_EX 用户提供力所能及的帮助。目前 C_TE_X 的主要工作包括维护 [C_TE_X 网站](http://www.ctex.org)，为用户提供学习交流的场所；开发中文 T_EX 系统安装配置软件；致力于 T_EX 文档的中文化工作。C_TE_X 的服务器和网络接入得到了中国科学院数学与系统科学研究院的支持。

2 C_TE_X 网站包括几个部分？

目前 C_TE_X 主要有三个部分：

- <http://www.ctex.org> 主页，发布最新软件、消息
- <http://bbs.ctex.org> 论坛，提问、讨论问题
- <ftp://ftp.ctex.org> FTP，提供软件资料下载、交流

3 这个 FAQ 的目的是什么？

本 FAQ 是为了收集 [C_TE_X 论坛](http://bbs.ctex.org) 上经常出现的问题以及解答，以便大家遇到问题时能方便迅速地找到帮助。本文档也适合初学者作为学习 T_EX 和 L^AT_EX 的参考资料。FAQ 中的解答力求正确完整，大部分都经过试验确认。

除了常见问题，本文档也收录一些难题和当前为止比较好的解决方法。如果你对 FAQ 中的问题有更好的方法，欢迎到 [C_TE_X 论坛](http://bbs.ctex.org) 上提出，或者联系本文档的维护人。在经过确认以后，你的方法将被收录到 FAQ 的新版本中。

4 我可以打印、传播、转载这份文档的内容吗？

可以。本文档采用 [自由软件组织](http://www.gnu.org/licenses/old/licenses.html) 颁布的 [GNU 自由文档许可证](http://www.gnu.org/licenses/old/licenses.html)。因此，你可以在保证文档的完整性前提下自由拷贝、传播这份文档。

你也可以摘录、转载这份文档中的部分内容，但是必须注明来源以及保证所有包含摘录内容的文档也都是自由文档，也就是可以免费得到的。详情请阅读 [GNU 自由文档许可证](http://www.gnu.org/licenses/old/licenses.html)。

B 基本概念

5 什么是 T_EX？

电子排版系统的出现给印刷出版业带来了一场革命，利用电子计算机及各种辅助设备，可以完成从文稿、图表的录入、编辑、修改、组版，直至得到各种不同用途、不同质量的输出结果。利用电子排版系统，可以减轻劳动强度，缩短出版周期。目前世界上有许多电子排版系统。这些系统各有特点，也各有自己的适用范围。T_EX 就是一种优秀的电子排版系统。

T_EX 提供了一套功能强大并且十分灵活的排版语言，它多达 900 多条指令，并且 T_EX 有宏功能，用户可以不断地定义自己适用的新命令来扩展 T_EX 系统的功能。许多人利用 T_EX 提供的宏定义功能对 T_EX 进行了二次开发，其中比较著名的有美国数学学会推荐的非常适合于数学家使用的 $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX 以及适合于一般文章、报告、书籍的 L^AT_EX 系统。

$\text{T}_{\text{E}}\text{X}$ 系统是公认的数学公式排得最好的系统。美国数学学会 (AMS) 鼓励数学家们使用 $\text{T}_{\text{E}}\text{X}$ 系统向它的期刊投稿。世界上许多一流的出版社如 Kluwer、Addison-Wesley、牛津大学出版社等也利用 $\text{T}_{\text{E}}\text{X}$ 系统出版书籍和期刊。

大部分的 $\text{T}_{\text{E}}\text{X}$ 系统都是免费的。Knuth 教授还公开了他的全部源程序。 $\text{T}_{\text{E}}\text{X}$ 系统目前已经在数百种计算机系统上得到实现。 $\text{T}_{\text{E}}\text{X}$ 系统的排版结果 DVI (DeVice Independent) 文件与输出设备无关。DVI 文件可以显示、打印、照排, 几乎可以在所有的输出设备上输出。 $\text{T}_{\text{E}}\text{X}$ 排版源文件及结果在各种计算机系统上互相兼容。

6 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 是什么?

Leslie Lamport 开发的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 是当今世界上最流行和使用最为广泛的 $\text{T}_{\text{E}}\text{X}$ 宏集。它构筑在 Plain $\text{T}_{\text{E}}\text{X}$ 的基础之上, 并加进了很多的功能以使得使用者可以更为方便的利用 $\text{T}_{\text{E}}\text{X}$ 的强大功能。使用 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 基本上不需要使用者自己设计命令和宏等, 因为 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 已经替你做好了。因此, 即使使用者并不是很了解 $\text{T}_{\text{E}}\text{X}$, 也可以在短短的时间内生成高质量的文档。对于生成复杂的数学公式, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 表现的更为出色。

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 自从八十年代初问世以来, 也在不断的发展。最初的正式版本为 2.09, 在经过几年的发展之后, 许多新的功能, 机制被引入到 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 中。在享受这些新功能带来的便利的同时, 它所伴随的副作用也开始显现, 这就是不兼容性。标准的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09, 引入了“新字体选择框架” (NFSS) 的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, $\text{S}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, $\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 等等, 相互之间并不兼容。这给使用者和维护者都带来很大的麻烦。为结束这中糟糕的状况, Frank Mittelbach 等人成立了 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}3$ 项目小组, 目标是建立一个最优的, 有效的, 统一的, 标准的命令集合。即得到 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 的一个新版本 3。这是一个长期目标, 向这个目标迈出第一步就是在 1994 年发布的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 。 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 采用了 NFSS 作为标准, 加入了很多新的功能, 同时还兼容旧的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ 。 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 每 6 个月更新一次, 修正发现的错误并加入一些新的功能。在 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}3$ 最终完成之前, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 将是标准的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 版本。

7 什么是 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}/\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$?

$\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}$ 是美国数学会提供的, 在 Plain $\text{T}_{\text{E}}\text{X}$ 基础上开发的 $\text{T}_{\text{E}}\text{X}$ 宏集。它主要用于排版含有很多数学符号和公式的科技类文章或书籍。 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}$ 给出了许多高级命令, 可以让使用者很方便地排版大型的, 复杂的数学公式。 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}$ 排版数学公式等的功能通过 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 中的宏包 `amsmath` 在 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 中得到实现。 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}$ 目前的版本为 2.1。

$\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 包括两部分, 一是上面提到的 `amsmath` 宏包, 主要的目的是用来排版数学符号和公式。另一部分是 `amscls`, 提供了美国数学会要求的论文和书籍的格式。 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 目前的版本为 2.0。

在提供 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-T}_{\text{E}}\text{X}$ 和 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 的同时, 美国数学会还提供一套数学符号的字库, `AMSFonts`。这套字库中增加了很多 $\text{T}_{\text{E}}\text{X}$ 的标准字库 Computer Modern 所没有的一些数学符号, 粗体数学符号等。`AMSFonts` 现在的版本为 2.2, 有 `Metafont` 和 `Type1` 两种字库提供下载。

8 $\text{T}_{\text{E}}\text{X}$ 和 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 是什么关系?

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 是 $\text{T}_{\text{E}}\text{X}$ 中的一种格式 (format), 是建立在 $\text{T}_{\text{E}}\text{X}$ 基础上的宏语言, 也就是说, 每一个 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 命令实际上最后都会被转换解释成几个甚至上百个 $\text{T}_{\text{E}}\text{X}$ 命令。但是, 普通用户可以无需知道这中间的复杂联系。就象编程的时候如果使用一些已经编译好的函数库和模板可以使我们仅仅用几个命令就实现很多功能一样, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 根据人们排版文章的习惯, 定义了许多命令和模板, 通过这些命令和模板, 我们可以很快的得到漂亮的排版结果。

9 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ 和 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 有什么区别?

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ 是 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 在 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 之前的一个版本 (参见问题 6)。 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 对 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ 做了很大的改进, 增加了很多新功能。从文件内容上看, 两者最显著的不同在于 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ 使用 `\documentstyle` 命令定义文档类型以及所包含宏包, 如

```
\documentstyle[twoside,epsfig]{article}
```

而 L^AT_EX 2_ε 使用 `\documentclass` 命令定义文档类型, 用 `\usepackage` 命令包含宏包, 如

```
\documentclass[twoside]{article}
\usepackage{epsfig}
```

如果你使用的不是几十年前的老机器, 你的系统的 L^AT_EX 都是 L^AT_EX 2_ε 版本。L^AT_EX 2.09 文件一般都可以在 L^AT_EX 2_ε 系统中以兼容方式编译。但是兼容方式编译速度慢, 而且很多 L^AT_EX 2_ε 的新功能无法使用。如果你不是需要编译以前的 L^AT_EX 2.09 文件, 你根本无需使用 L^AT_EX 2.09, 也不用知道 L^AT_EX 2.09 与 L^AT_EX 2_ε 的差别。大部分 L^AT_EX 2.09 文件都可以通过用 `\documentclass` 命令和 `\usepackage` 命令代替 `\documentstyle` 命令修改为 L^AT_EX 2_ε 格式。有时可能需要一些特殊宏包, 例如 `latexsym`, 对旧的 L^AT_EX 2.09 命令提供支持。

10 MiK_TE_X、f_pT_EX、t_eT_EX、C_TE_X 又是什么关系?

T_EX 在不同的硬件和操作系统上有不同的实现版本。这就像 C 语言, 在不同的操作系统中有不同的编译系统, 例如 Linux 下的 gcc, Windows 下的 Visual C++ 等。有时, 一种操作系统里也会有好几种的 T_EX 系统。目前常见的 Unix/Linux 下的 T_EX 系统是 t_eT_EX, Windows 下则有 MiK_TE_X 和 f_pT_EX。C_TE_X 指的是 C_TE_X 中文套件的简称, 是把 MiK_TE_X 和一些常用的相关工具, 如 GSview, WinEdt 等包装在一起制作的一个简易安装程序, 并对其中的中文支持部分进行了配置, 使得安装后马上就可以使用中文。

11 什么是 CTAN ?

CTAN 是英文 Comprehensive T_EX Archive Network 的首字母缩写, 直译成中文就是综合 T_EX 文件网络, 它包含有三个主服务器和几十个镜像服务器, 以 FTP 和 WWW 方式向所有的 T_EX 用户提供文件服务。CTAN 可以说几乎收罗了世界上所有的 T_EX 相关软件、文档, 并保证不断更新。你可以访问

<http://www.ctan.org>

来得到更详细的信息。在 CTAN 中搜索文件可以参见 问题 14。

C 安装升级

12 哪里可以下载 C_TE_X 中文套装软件?

C_TE_X 首页上的下载中心:

<http://www.ctex.org/modules.php?op=modload&name=Downloads&file=index>

或者直接到 C_TE_X 的 FTP 服务器上下载:

<ftp://ftp.ctex.org/pub/tex/systems/ctex/>

13 哪里可以下载 MiK_TE_X 的最新版本?

可以到 MiK_TE_X 的主页:

<http://www.miktex.org>

或者 C_TE_X 提供的 CTAN 镜像中下载:

<ftp://ftp.ctex.org/CTAN/systems/win32/miktex/setup/setup.exe>

然后执行 setup.exe, 在其提供的站点列表中选择合适的网站 (C_TE_X FTP 是列表中唯一的中国大陆网站), 按照程序提示即可完成下载。

14 哪里能找到我想要的宏包？

CTAN（见问题 11）可以说几乎收罗了世界上绝大多数的 T_EX 相关软件、文档，并保持更新，为广大用户提供了方便。虽然 CTAN 的目录结构很有规则，并不算太难找，但对于许多对其目录结构不熟悉的人来说，还是一个问题。特别是很多初学者，经常找不到想要的东西。

CTAN 提供了一个搜索引擎

<http://www.ctan.org/search/?action=/index.html>

通过它，你可以确定你需要的文件在 CTAN 目录中的相对位置，然后就可以在任何一个 CTAN 镜像中下载了。CTAN 在全世界有几十个镜像服务器，每日同步更新，本站的 FTP 也是其中之一。你可以从搜索引擎提供的列表中挑选一个离你最近服务器下载。

注意一点：不同的镜像服务器存放 CTAN 的根目录可能不同。例如，CT_EX FTP 是在

/CTAN/

目录下。把搜索引擎的结果加到这个目录后面，就是这个文件在我们的 FTP 上的路径了。

15 如何升级现有的 MiK_TE_X？

无论是 CT_EX 中文套装包含的 MiK_TE_X 还是单独安装的 MiK_TE_X 都可以使用以下方法升级（只适用于 MiK_TE_X 2.2 以后版本）运行开始菜单中 MiK_TE_X 组中的 MiK_TE_X Update Wizard，在其中提供的站点列表中选择合适的网站，则程序会自动完成下载所需的文件并升级。也可以使用安装时的程序 setup.exe 把最新的内容下载到你的硬盘上以后，再运行 MiK_TE_X Update Wizard 并选择从本地 (local folder) 升级。

16 我下载了一些新的宏包，该如何安装到我的系统中去？

如果你得到的宏包已经是编译过的，也就是已经有 .sty 或者 .cls 文件，你只需把这些文件放在

```
localtexmf\tex\latex\
```

下的某个子目录中，可以自己建一个。然后你还要让 T_EX 系统知道这个新的宏包的存在，你需要刷新系统的文件名数据库（见问题 23）。现在你可以使用这个宏包了。

你也可以把这些宏包文件和你的 T_EX 源文件放在同一目录下，这种情况下你不需要刷新系统的文件名数据库，但只有同一个目录下的文件可以使用这些宏包。

如果你拿到的是宏包的源文件，一般来说都包含两个文件：一个扩展名为 .ins，另一个扩展名为 .dtx。此外，通常会有一个 readme.txt 对宏包进行简要的说明。你应该先阅读这个文件，接着对 .ins 文件运行 L^AT_EX 命令，这会产生一个或者数个 .sty 文件以及其它相关文件。然后参照上面的方法结合宏包说明文件的要求，把生成的文件加入你的 T_EX 系统中。你还可以直接对 .dtx 文件运行 L^AT_EX 从中产生宏包的帮助文档。更详细的说明可以参见 lshort [4]。

17 CT_EX 中文套装软件的 Basic、Pro、Full 版本都有什么区别？

CT_EX 根据不同用户的要求制作了 Basic、Pro、Full 三个不同大小版本的 CT_EX 中文套装。他们之间的区别仅在于其中的 MiK_TE_X 包含的内容。Basic 版本中的 MiK_TE_X 只有最基本的 L^AT_EX 系统。如果你只是想了解试用一下 L^AT_EX 系统而不想安装太多东西，那么就可以安装这个版本，编译一些简单的 L^AT_EX 文件已经足够了。或者你想自己定制安装所需的 L^AT_EX 组件和宏包，你也可以安装这个版本，然后加入需要的组件和宏包。对于因为硬盘空间有限而无法安装 Pro 版本和 Full 版本的用户，也可以采用这个方法来解决。Basic 版本所需的安装空间为一百多兆。

New
v0.4

Full 版本中包含的是完整的 MiKTeX，适用于需要很多功能而且有较大硬盘空间的用户。Full 版本需要六百多兆的安装空间。

Pro 版本是经过精心定制的，大小介于 Basic 版本和 Full 版本中间。其中包含了常用的 MiKTeX 内容，适用于绝大多数的中国用户。相对于 Full 版本，Pro 版本可以节省一百多兆的安装空间。

18 CTeX-Fonts、CTeX-CS、CTeX-Ext 都是干什么的？和 CTeX 中文套装有什么关系？

New
v0.4

除了 CTeX 中文套装，CTeX 还制作了几个相关的安装包，分别是 CTeX-Fonts、CTeX-CS 和 CTeX-Ext。CTeX-Fonts 中包含了几种常用的中文字体的 Type1 字库和相应的配置文件。在没有安装 CTeX-Fonts 时，CTeX 缺省使用 Windows 系统中的 TrueType 字库。关于两种字库的差别，参见问题 31 和问题 32。此外，对于 Windows 2000 以及之后的 Windows 操作系统，安装 CTeX-Fonts 还可以避开其中宋体字库的问题（参见问题 82）。目前包含的字体有宋体、仿宋体、楷体、黑体、隶书和幼圆六种。

CTeX-CS 是 CTeX 套装中的中文安装部分，包括 CJK 和 CCT 的所有文件。对于使用 CTeX 套装的用户，CTeX-CS 可以用于将中文相关的设置恢复到 CTeX 套装的初始状态。对于没有安装 CTeX 套装的用户，则可以在用户自行安装的 MiKTeX 中增加中文支持。此外，CTeX-CS 还支持为 fpTeX（即 TeXLive）增加中文支持。

CTeX-Ext 是 CTeX 套装的扩展部分，用于一些 TeX 相关工具的中文设置和安装。目前的版本包括 ConTeXt 的中文设置安装。

D 新手入门

19 我是新手，该如何开始？

首先，我们建议你至少阅读一些简单的教材，例如 lshort 中文版^[4]，对 L^AT_EX 有一个基本的了解。

在安装系统方面，如果你不需要中文支持，可以安装 MiKTeX 或者 fpTeX，如果需要用到中文，建议使用 CTeX 中文套装来学习使用。否则，光是中文的设置安装就可能使你焦头烂额。在 Linux 下，你可以使用 teTeX。

在安装完系统以后，先输入一个最简单的例子（英文的例子参见问题 20，中文的例子参见问题 25 和问题 26）来看看系统是否正确安装了。之后你就可以开始使用了。

如果遇到问题，建议使用一个尽可能简单的例子来测试你的问题，否则的话，在一个复杂文件里排除错误会花掉你很多的精力。而且由于大的文档上下文的命令互相影响，会使得调试变得困难，特别是对初学者来说。

如果你使用的是别人的模板，那么遇到问题最好的解决办法就是问模板的作者，请求他的帮助。如果是一般的问题，你可以到 [CTeX 论坛](#) 或者其他国内外的 TeX 网站上请求帮助。在 [CTeX 网站](#) 上你可以找到这些网络链接。

20 能否举一个简单的英文排版例子？

Revised
v0.4

```
\documentclass{article}
\begin{document}
This is a test file.
\end{document}
```

这个例子需要保存为 .tex 后缀的文件，然后用 L^AT_EX 编译，生成的 DVI 文件可以用 DVI 浏览器进行查看或者用 dvips 转换成 PostScript 文件。假设文件名是 test.tex，生成 PostScript 文件的完整编译过程是：

```
latex test
dvips test
```

21 用什么编辑工具来编辑 L^AT_EX 文件?

L^AT_EX 文件是标准的文本文件，任何一个文本编辑工具都可以编辑 L^AT_EX 文件。但是有些编辑器针对 L^AT_EX 文件作了设置和优化，例如增加可以直接调用 L^AT_EX 编译程序的菜单或者按钮，使得你可以更方便的进行编辑、编译工作。

在 Windows 中对 L^AT_EX 支持最好的是 WinEdt，针对 MiK_TE_X 和 fp_TE_X 都有预制好的菜单和工具条按钮，安装完就可以使用。WinEdt 还可以通过工具条输入许多数学符号，免去了记忆的烦恼。在各种细节上，例如项目管理、彩色关键字、括号匹配等等，WinEdt 对 L^AT_EX 的支持都很完善，你可以在使用中慢慢发掘他的强大功能。还有一部分用户喜欢使用 UltraEdit 和 Editplus，这两个都是功能强大的通用编辑工具，你可以针对 L^AT_EX 定制菜单和工具条。缺点就是这些都要你自己动手，在一些细的方面，也没有 WinEdt 考虑得周到，毕竟不是专门针对 L^AT_EX 的编辑器。此外，还有一些其他的编辑器，如 WinTeX、TeXnicCenter 等，如果你感兴趣，可以一一尝试，挑一个自己喜欢的来编辑 L^AT_EX 文件。

在 Linux/Unix 下，我们推荐使用 vi 或者 Emacs 作为 L^AT_EX 编辑器。

22 什么是导言区 (preamble) ?

L^AT_EX 文档的标准结构是

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

在 `\begin{document}` 和 `\end{document}` 之间的是我们的正文部分。而在 `\documentclass{article}` 和 `\begin{document}` 之间的部分，通常用于定义一些全局命令和参数，这个部分就称为导言区，英文是 *preamble*。在 `\end{document}` 之后的部分会被 L^AT_EX 完全忽略，可以用来写一些关于文档的注释，或者暂时不想包含到正文中去的内容，也有人把它叫做 *postamble*，因为它和导言区 (*preamble*) 处在完全相对的位置。

23 什么是文件名数据库? 怎么刷新?

T_EX 系统通过这个数据库记录了所需要的各种文件的名称和具体目录位置。每次你往系统中添加文件后，都要记得刷新这个数据库，否则 T_EX 有可能找不到需要的文件。

刷新文件名数据库的具体命令取决于你使用的 T_EX 系统。te_TE_X 和 fp_TE_X 的命令是

```
texhash
```

MiK_TE_X 用户可以在 Dos 窗口中输入以下命令

```
initexmf --update-fndb
```

或者使用 MiK_TE_X 提供的图形界面工具 MiK_TE_X Options。在开始菜单中找到 MiK_TE_X Options，打开点击 General 页上的 Refresh Now 按钮即可。

E 中文支持

24 比较一下 CJK 与 CCT 的优缺点？

CJK 比 CCT 的优越之处有以下几点：

1. 字体定义采用 L^AT_EX NFSS 标准，生成的 DVI 文件不必像 CCT 那样需要用 patchdvi 处理后才能预览和打印。而且一般 GB 编码的文件也不必进行预处理就可直接用 L^AT_EX 命令进行编译。
2. 可使用多种 TrueType 字体和 Type1 字体，生成的 PDF 文件更清楚、漂亮。
3. 能同时在文章中使用多种编码的文字，如中文简体、繁体、日文、韩文等。

当然，CCT 在一些细节上，如字体可用中文字号，字距、段首缩进等更符合中文的习惯。而 CJK 就没有针对中文习惯指定特殊的排版标准。

目前，CCT 的作者已经推出了最新版的 CCT，可以使用 CJK 的字库，从而也避免了预处理和 patchdvi 的过程，使用更方便，排版效果更美观了。最新版的 CCT 可以在

<ftp://ftp.cc.ac.cn/pub/cct/>

下载，或者使用 C_T_EX 提供的工具 C_T_EX-CS 升级。

25 能否举一个简单的 CCT 中文排版例子？

一个简单的老版本 CCT 格式的例子是：

```
\documentclass{cctart}
\begin{document}
\kaishu 这是中文楷体字。
\end{document}
```

这个例子需要保存为 .ctx 后缀的文件，然后用 cct 命令进行预处理，生成同名 .tex 文件。再用 L^AT_EX 编译，生成的 DVI 文件需要 patchdvi 进行处理后才能用 DVI 浏览器进行查看或者用 dvips 转换成 PostScript 文件。假设文件名是 test.ctx，完整的编译过程是：

```
cct test
latex test
patchdvi -r600x600 -b test.dvi temp.dvi
del test.dvi
ren temp.dvi test.dvi
dvips test
```

新版的 CCT 除了保留原来的处理方式以外，增加了两种新的处理方式。第一种是用 T_EX 的处理来代替原来的 cct.exe 的预处理。这种方式的源文件和老的文件相同，但是不用再存成 .ctx 为后缀的文件，也不用 cct 命令进行预处理。除此之外，其余和老的处理方式基本相同。

第二种处理方式是采用 CJK 的中文字库，需要在系统中安装好 CJK 字库。与第一种方式的主要区别就在于去掉了 patchdvi 处理 DVI 文件的需要。使用上，是在 \documentclass 命令中加上参数 CJK。具体的例子如下：

```
\documentclass[CJK]{cctart}
\begin{document}
\kaishu
这是中文楷体字。
\end{document}
```

这个例子可以象英文文档一样的编译（见 问题 20）得到正确的输出。

26 能否举一个简单的 CJK 中文排版例子？

在安装好 CJK 的系统中，下面这个例子可以象英文文档一样的编译（见 问题 20）得到正确的输出。

```
\documentclass{article}
\usepackage{CJK}
\begin{document}
\begin{CJK*}{GBK}{kai}
这是中文楷体字。
\end{CJK*}
\end{document}
```

27 在 CJK 中如何使段首缩进两个中文字符宽度？

在 CJK 环境中使用命令

```
\CJKindent
```

28 在 CJK 中如何调整汉字的间距？

CJK 会在两个汉字之间插入一个 `\CJKglue`，因此我们修改 `\CJKglue` 的定义就可以调整汉字的间距。用法是

```
\renewcommand{\CJKglue}{\hspace <b> plus <p> minus <m>}
```

其中 $\langle b \rangle$ 、 $\langle p \rangle$ 和 $\langle m \rangle$ 都是 $\text{T}_{\text{E}}\text{X}$ 长度，如 0.05pt。执行以上命令后，汉字的标准间距就是 $\langle b \rangle$ ，在需要的时候向上向下调整，调整幅度分别不超过 $\langle p \rangle$ 和 $\langle m \rangle$ 。

CJK 的缺省值是 `\hspace 0pt plus 0.08\baselineskip`。

29 CJK 和 CJK* 有何不同？

CJK 宏包有两种不同的处理方式。一个是

```
\begin{CJK}...
...
\end{CJK}
```

称为 CJK 模式；另一个是

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

称为 **CJK*** 模式。两个模式的区别在于 **CJK*** 会忽略 **CJK** 字符之间的空格，这是我们中文的习惯。而 **CJK** 则使用英文的习惯，即词之间保留空格，当然如果空格多于一个，**T_EX** 也会忽略多余的空格。

两种方式各有利弊。首先来看 **CJK** 模式，它的好处是符合 **T_EX** 的英文处理习惯，无需对空格进行额外处理，坏处是不符合中文习惯。一个经常发生的问题就是，由于 **T_EX** 将分行符也视为空格，因此就造成中文字符间产生多余的空格，使得排版结果很难看。解决方法是在行尾加上注释字符 `%` 来告诉 **T_EX** 忽略两行间的分行符。因此，我们不建议使用 **CJK** 模式。

CJK* 模式由于自动忽略中文字符之间的所有空格，因此没有上面的这种副作用。但是这也带来了另一方面的问题，就是如果想在中文字符中间加入空格就必须加以保护，避免被忽略。加保护的方法是在空格前面加上 `\` 字符。实际上 **CJK*** 模式忽略中文字符后面的空格，因此中文后面如果接着英文，必须加上保护的空格 `\` 或者 `~` 符号，否则可能造成断行错误（参见 问题 30）。

两种模式还可以通过命令互相转换，使用

```
\CJKnospace
```

则进入 **CJK*** 模式，用

```
\CJKspace
```

则进入 **CJK** 模式。

30 中英文混排时为何无法正确分行？

首先，英文标点符号后要留空格，这是英文的习惯，否则 **T_EX** 无法断句。

其次，要用空格隔开中英文字符，**T_EX** 才能正确分行。

但是如果是用正常的空格，由于空格间隙太大，容易造成比较难看的排版效果。因此，**CJK** 宏包重新定义了符号 `~` 的意义。`~` 的原始定义是不可断行的空格，**CJK** 将其重定义成一个比较合适的弹性距离，大约相当于四分之一一个汉字宽度，用于分隔中英文字符。我们通过使用命令

```
\CJKtilde
```

来实现 `~` 的重定义。如果要恢复原始定义，可以用命令

```
\standardtilde
```

注意：在 **CJK*** 模式下中文字符后面的空格会被忽略，因此中文字符后面必须加上保护的空格 `\` 或者 `~` 符号来隔开英文字符。

31 什么是点阵字库和矢量字库？

我们都知道，各种字符在电脑屏幕上都是以一些点来表示的，因此也叫点阵。最早的字库就是直接把这些点存储起来，就是点阵字库。常见的汉字点阵字库有 16x16，24x24 等。点阵字库也有很多种，主要区别在于其中存储编码的方式不同。点阵字库的最大缺点就是它是固定分辨率的，也就是每种字库都有固定的大小尺寸，在原始尺寸下使用，效果很好，但如果将其放大或缩小使用，效果就很糟糕了，就会出现我们通常说的锯齿现象。因为需要的字体大小组合有无数种，我们也不可能为每种大小都定义一个点阵字库。于是就出现了矢量字库。

矢量字库是把每个字符的笔划分解成各种直线和曲线，然后记下这些直线和曲线的参数，在显示的时候，再根据具体的尺寸大小，画出这些线条，就还原了原来的字符。它的好处就是可以随意放大缩小而不失真。而且所需存储量和字符大小无关。矢量字库有很多种，区别在于他们采用的不同数学模型来描述组成字符的线条。常见的矢量字库有 Type1 字库和 TrueType 字库。

32 在 CJK 中都可以使用哪些矢量字库？

在 CJK 中可以使用 Type1 字库和 TrueType 字库，都能产生很好效果的输出，但有许多细微的差别，输出结果取决于所用的工具。下面根据转换工具和输出文件类型来比较它们的差异。

1. 使用 dvips 生成 PostScript 文件
 - dvips 可以直接使用 Type1 字库，生成的 PostScript 文件中包含了字符的矢量描述，具有很好的缩放效果。
 - dvips 不能直接使用 TrueType 字库，因此必须通过 ttf2pk 生成中间的pk字库，这是一种点阵字库，所以生成的 PostScript 文件在效果较差。
2. 使用 dvipdfm 生成 PDF 文件和 dvips 类似，可以直接使用 Type1 字库或者通过 ttf2pk 间接使用 TrueType 字库。
3. 使用 pdfL^AT_EX 生成 PDF 文件可以直接使用两种字库，因此都可以产生效果很好的输出。
4. 使用 dvipdfmx 生成 PDF 文件可以直接使用两种字库，因此都可以产生效果很好的输出。而且直接使用 TrueType 字库时生成的 PDF 文件是唯一能支持对中文内容进行正常拷贝粘贴的。

33 两种类型矢量字库会不会发生冲突？

不会，每个工具都有自己的 config 和 map 文件按一定的顺序查询有哪些字库可以使用，来决定使用哪种字库。

34 如何生成支持中文拷贝粘贴的 PDF 文件？

生成可以拷贝粘贴中文的 PDF 文件目前只有 dvipdfmx（原名 dvipdfm-cjk）可以做到，它必须配置成直接使用中文 TrueType 字库。

F 文档排版

35 如何使首段段首也缩进？

使用 indentfirst 宏包：

```
\usepackage{indentfirst}
```

36 如何设置双倍或者更大的行间距？

使用命令

```
\renewcommand{\baselinestretch}{\langle倍数\rangle}
```

37 如何注释掉一大段内容，不做编译？

用命令

```
\iffalse
```

和

```
\fi
```

把需要注释的内容包含起来，这样 L^AT_EX 编译的时候就会忽略这些内容，不做编译。

也可以使用 verbatim 宏包提供的 comment 环境：

```
...
\usepackage{verbatim}
...
\begin{comment}
This is comment.
\end{comment}
...
```

38 如何生成分栏的文档？

Revised
v0.4

如果全文都是双列分栏格式，可以在 `\documentclass` 命令中加入参数 `twocolumn`。例如：

```
\documentclass[twocolumn]{article}
```

如果要生成单双列混合的分栏格式，或者超过两列的分栏格式，则要用到 multicol 宏包的 `multicols` 环境。

```
\usepackage{multicol}
```

然后将需要多列的地方使用：

```
\begin{multicols}{<列数>}
```

和

```
\end{multicols}
```

包含起来。

39 如何使得双栏文档保持底部对齐？

Revised
v0.4

在 L^AT_EX 标准文档类中使用 `twocolumn` 选项产生双栏格式文档的时候，如果一页的内容不足以排满页面时，L^AT_EX 的结果是先排满左栏，然后再排版右栏。这样就造成了两栏的长度不相等，底部不齐。很多时候我们不希望这种排版结果。

这个问题只有在用标准文档类的 `twocolumn` 模式的时候才会遇到，使用 `multicol` 宏包就可以避免这个问题（参见问题 38）。但是 `multicol` 宏包也有一些缺点，例如不允许单栏的浮动对象（参见问题 58）。

要使标准文档类的 `twocolumn` 模式下也能做到双栏底部对齐，可以通过手工调整分栏的位置。在 `twocolumn` 模式下 `\newpage` 命令使 L^AT_EX 开始在新的一栏上排版，如果单前栏是左栏，就换到右栏上开始排版。在合适的地方使用 `\newpage` 命令可以使得双栏底部对齐（注意，不能用 `\clearpage` 命令，这个命令会开始在新的一页上排版，而不管当前在哪一栏上）。几乎没有人愿意使用这种方法，不仅因为它需要你手工计算需要插入命令的位置，每次改动文档都有可能需要变动 `\newpage` 命令的位置，而且对于一些自动生成的文档内容（例如参考文献和索引），要插入 `\newpage` 命令十分困难。

`flushend` 宏包通过修改 L^AT_EX 核心定义来解决这个问题。使用它提供的 `\flushend` 命令可以使得双栏底部对齐，而 `\raggedend` 命令则取消底部对齐。`\flushend` 命令会在宏包装入的时候自动运行。`flushend` 宏包只对文档最后一页有效，也就是说所有中间的强制换页，或者章节的自动换页导致的双栏底部不齐现象都无法解决。

`balance` 是另一个针对该问题的宏包，它比 `flushend` 宏包更加完善，而且对整个文档的任何位置都有效。`balance` 宏包提供了 `\balance` 命令来打开双栏底部对齐机制，`\nobalance` 命令来关闭这个机制。和 `flushend` 宏包不同，它在装入的时候缺省不打开对齐机制，因此需要在文档中加入 `\balance` 命令来启用底部对齐。

在使用这些宏包的时候需要注意，单栏的浮动对象有可能破坏这些宏包的对齐机制。这种时候需要手工调整浮动对象的位置来改善排版效果。这也是 `multicol` 宏包不支持单栏浮动对象的原因之一。

40 如何在双栏文档中使用通栏的摘要？

在排版双栏文章的时候最经常遇到的一个问题就是如何产生通栏（不分栏）的摘要。解决这个问题的方法是：

```
\documentclass[twocolumn]{article}
...
\begin{document}
...
\title{...}
\author{...}
\twocolumn[
  \begin{@twocolumnfalse}
    \maketitle
    \begin{abstract}
      ...
    \end{abstract}
  \end{@twocolumnfalse}
]
...
\end{document}
```

注意，`\maketitle` 命令必须放在示例中的位置，否则标题结束后会另起一页。但是标题和作者的定义可以放在前面的位置。

上面这个方法会带来一个副作用，那就是 `\thanks` 命令在 `\title` 和 `\author` 中失去了作用，无法产生脚注。这个问题可以通过附加一小段代码来解决：

```
...
\title{...}
\author{...\thanks{}} % 这里用一个空的 \thanks
\twocolumn[
  ...
]
{
  \renewcommand{\thefootnote}{\fnsymbol{footnote}} %如果使用数字做脚注标号，则不需要这一行
  \footnotetext[1]{真正的脚注内容}
}
...
```

`abstract` 宏包提供了一个 `\saythanks` 命令和 `onocolabstract` 环境来简化这个稍嫌复杂的过程。它们的用法是：


```

...
\title{...}
\author{... \thanks{...}}
\twocolumn[
  \maketitle
  \begin{onecolabstract}
    ...
  \end{onecolabstract}
]
\saythanks
...

```

41 如何产生全部横排的文档?

使用文档类的 `landscape` 参数, 例如

```
\documentclass[landscape]{article}
```

为了产生正确的结果, `dvips` 必须加上 `-t landscape` 参数, `dvipdfm/dvipdfmx` 必须加上 `-l` 参数。

42 如何使文档中的某一页横排?

使用 `lscape` 宏包提供的 `landscape` 环境, 例如

```

...
\usepackage{lscape}
...
\begin{landscape}
This is a landscape page.
\end{landscape}
...

```

将产生一个横排的页面, 但是只有在 PostScript 或者 PDF 文件中才能看到正确的效果。

43 如何使用 New Time Roman 字体?

用 `times` 宏包就可以把文本的缺省字体换成 Times New Roman 字体。如果你安装有 MathTime 字库, 那么还可以使用 `mathtime` 宏包来把数学环境中的字体都变成 Times New Roman 字体。例如

```

\usepackage{times}
\usepackage[mtbold,mtpluscal,mtplusscr]{mathtime}

```

Revised
v0.4

G 数学公式

G.1 编号

44 如何实现定理、定义、推论等的统一编号?

现在很多文章的定理、定义、推论、例子都采用统一编号, 不再是独立编号。如定义 1.1, 接下来可能是定理 1.2, 然后推论 1.3, 等等。这可以用如下的定义来完成:

```

\newtheorem{thm}{Theorem}[section] %如果不采用章节号做前缀, 则不用[section]
\newtheorem{defn}[thm]{Definition} %这句定义使得defn环境和thm共享编号
\newtheorem{lem}[thm]{Lemma} %这句定义使得lem环境和thm共享编号

```

45 如何让公式编号中使用当前章节编号做前缀？

如果你想让 L^AT_EX 产生的公式编号中包含当前的章节编号，例如第一节的公式编号是 (1.1) (1.2)，第二节的公式编号是 (2.1) (2.2) 等等，你可以重新定义 `\theequation`：

```
\renewcommand{\theequation}{\thesection.\arabic{equation}}
```

这么做有一个小问题，每当新的章节开始的时候，公式编号不会重新从 1 开始。你不得不使用 `\setcounter` 在新的章节开始的时候把公式编号的计数器清零。

`amsmath` 宏包提供了一个 `\numberwithin` 命令来自动完成这些工作。

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

使得公式编号和章节编号联系在一起，而且章节编号改变时公式编号自动清零。`\numberwithin` 命令不光可以用于修改公式编号的格式，也可以用于将任意的两个计数器联系在一起。

46 如何实现如 (1a) (1b) 这种公式编号？

使用 `amsmath` 宏包中的 `subequations` 环境。例如：

```
\begin{subequations}
\begin{equation}
A = B
\end{equation}
\begin{equation}
C = D
\end{equation}
\end{subequations}
```

$A = B$	(1a)
$C = D$	(1b)

所有在同一个 `subequations` 环境中的公式共享一个主编号（例子中的 1）。

如果想用数字代替字母作为公式的子编号，或者修改编号的格式，可以在 `subequations` 环境内部重新定义命令 `\theequation`，例如

```
\begin{subequations}
\renewcommand{\theequation}
{\theparentequation-\arabic{equation}}
\begin{equation}
A = B
\end{equation}
\begin{equation}
C = D
\end{equation}
\end{subequations}
```

$A = B$	(2-1)
$C = D$	(2-2)

公式的主编号使用计数器 `parentequation`，子编号使用计数器 `equation`。如果这种公式很多，在每一个 `subequations` 中都加上这么一句命令不是一个好的方法。你可以重新定义一个新的环境：

```
\newenvironment{mysubequations}
{\begin{subequations}\renewcommand{\theequation}{\theparentequation-\arabic{equation}}}
{\end{subequations}}
```

然后使用自定义的环境代替 `subequations` 就可以了。

47 如何让一个公式编号表示为另一个公式编号的变化形式？

我们经常使用一个公式编号加上某些符号如 ' 来表示公式的一个变化。这样的公式编号可以通过 `amsmath` 宏包的 `\tag` 命令加上 L^AT_EX 的引用机制来实现。例如

```
\begin{equation} \label{eq:original}
A = B
\end{equation}
...
\begin{equation}
\tag{\ref{eq:original}$'$} \label{eq:origprime}
A = B + 1
\end{equation}
...~公式~\ref{eq:origprime}~是由
公式~\ref{eq:original}~变化而来~...
```

$$A = B \quad (3)$$

...

$$A = B + 1 \quad (3')$$

... 公式 3' 是由公式 3 变化而来 ...

类似的，你也可以实现其他形式的编号变化。

G.2 字体

48 实数域 \mathbb{R} 或复数域 \mathbb{C} 等的字体该用什么命令？

使用 `amsmath` 宏包提供的

$$\mathbb{\langle 字母 \rangle}$$

命令，例如：

```
 $x \in \mathbb{R}$  and  $c \in \mathbb{C}$ 
```

$$x \in \mathbb{R} \text{ and } c \in \mathbb{C}$$

49 书中向量常用斜体加黑体表示，该用什么字体命令排版？

使用 `bm` 宏包提供的

$$\mathbf{\langle 字母 \rangle}$$

命令，例如：

```
 $\mathbf{x}^T \mathbf{y}$ 
```

$$\mathbf{x}^T \mathbf{y}$$

G.3 其它

50 如何使上下限出现在求和、积分符号的上下方而不是右边？

数学公式中求和、积分符号的上下限的位置取决于是在行间公式还是独立公式。在行间公式中，类似的上下限出现在符号的右边，而在独立公式中出现在符号的上下方。这一规则可以通过命令

$$\lim_{\langle 字母 \rangle}$$

和

```
\nolimits
```

来改变。行间公式缺省使用 `\nolimits` 而独立公式缺省使用 `\limits`。如果要改变缺省规则，就在符号后面，上下限之前加入上面的两个命令之一。例如缺省情况是

```
$$\sum_{i=1}^n i^2 = 0$$
If $\sum_{i=1}^n i^2 = 0$ then ...
```

$$\sum_{i=1}^n i^2 = 0$$

If $\sum_{i=1}^n i^2 = 0$ then ...

我们可以修改为

```
$$\sum\nolimits_{i=1}^n i^2 = 0$$
If $\sum\limits_{i=1}^n i^2 = 0$ then ...
```

$$\sum_{i=1}^n i^2 = 0$$

If $\sum_{i=1}^n i^2 = 0$ then ...

51 如何实现上下方能输入公式文字且可伸长的等号或者箭头符号？

宏包 `amsmath` 提供了两个可以伸长的单箭头符号

```
\xleftarrow[<下方公式>]{<上方公式>}
```

和

```
\xrightarrow[<下方公式>]{<上方公式>}
```

示例如下：

```
$ A \xleftarrow{n=0} B \xrightarrow[T]{n>0} C $
```

$$A \xleftarrow{n=0} B \xrightarrow[T]{n>0} C$$



至于其他样式的符号，我们可以利用 `amsmath` 提供的一些内部命令定义几个通用的 `ExtendSymbol` 命令来实现：

```
...
\usepackage{amsmath}
...
\makeatletter
\def\ExtendSymbol#1#2#3#4#5{\ext@arrow 0099{\arrowfill@#1#2#3}{#4}{#5}}
\def\RightExtendSymbol#1#2#3#4#5{\ext@arrow 0359{\arrowfill@#1#2#3}{#4}{#5}}
\def\LeftExtendSymbol#1#2#3#4#5{\ext@arrow 6095{\arrowfill@#1#2#3}{#4}{#5}}
\makeatother
```

三个命令的差别只在于符号上下的字符位置处理上。简单解释一下五个参数的意义：前面三个参数用来组合生成最后的符号，第一个参数代表左端的符号，第二个参数的代表中间重复的符号，第三个参数代表右端的符号。最后两个参数分别是符号下的字符串和符号上的字符串。现在我们就可以利用 `ExtendSymbol` 来定义我们想要的符号了。例如，定义一个长的双箭头

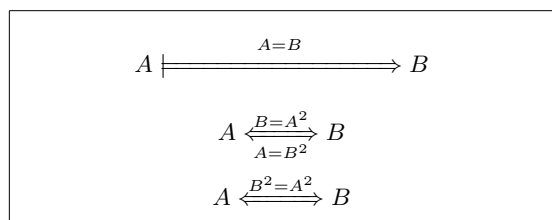
```
\newcommand\myRightarrow[2] []{\RightExtendSymbol{|}{=}{\Rightarrow}{#1}{#2}}
```

又如定义一个长的双向箭头

```
\newcommand\myArrow[2] []{\ExtendSymbol{\Leftarrow}={}\Rightarrow}{#1}{#2}}
```

使用的时候可以用宽度为 0 的 rule 支撑指定符号的最小长度也可以根据符号上下的字符长度来确定符号长度

```
$$ A \myRightarrow[\rule{3cm}{0cm}]{A=B} B $$
$$ A \myArrow[A=B^2]{B=A^2} B $$
$$ A \myArrow[B^2=A^2] B $$
```



大家可以举一反三。

H 图形表格

H.1 图形

52 L^AT_EX 如何使用图形?

由于 DVI 文件经常被转为 PostScript 文件，所以 L^AT_EX 支持最好的是 EPS 格式（Encapsulated PostScript，是 PostScript 语言的子集）的图形。目前的 L^AT_EX 2_ε 包含两个功能相同的图形宏包：graphics 和 graphicx 宏包。两者的差异是 graphics 宏包使用标准的 L^AT_EX 参数语法规则，而 graphicx 版本则使用更为简便灵活的命名机制。

一个使用图形的简单例子是：

```
\documentclass{article}
\usepackage{graphicx}
\begin{document}
\includegraphics{file.eps}
\end{document}
```

因为 `\includegraphics` 不会结束当前段落，所以它能够在文本行中放置图形。如果需要放置独立的图形，则需要用到 figure 浮动环境。例如：

```
\documentclass{article}
\usepackage{graphicx}
\begin{document}
\begin{figure}
\includegraphics{file.eps}
\caption{This is caption of figure}
\end{figure}
\end{document}
```

在 figure 环境中可以使用 `\caption` 命令来制定图形的标题，这个标题将会出现在图形的上方或者下方，还会通过使用 `\listoffigures` 命令出现在一个图形目录中。

我们建议 L^AT_EX 用户详细阅读《L^AT_EX 图形指南》^[3] 一书，可以使得你对在 L^AT_EX 中使用图形有一个全面的了解，避免走弯路。

53 L^AT_EX 支持哪些图形格式？

目前 L^AT_EX 只有 EPS 格式的图形在所有的 TeX 系统中都能得到全面的支持。而其他的图形，如 JPEG 可以在某些系统中使用，这取决于所用的 DVI 转换程序。因为一般最终要得到的都是 PostScript 或者 PDF 格式的文件，因此先将图形转换为 EPS 格式仍然是最有效、最方便的图形处理方法。



虽然如此，我们在这里还是给出一个直接使用 JPEG 图形和 PNG 图形的方法。假设我们有 `ctex-j.jpg` 和 `ctex-p.png` 图形，以及两个格式为

```
%%BoundingBox: 0 0 width height
```

的 `ctex-j.bb` 和 `ctex-p.bb` 文件，其中的 `width` 为对应图形的宽度，`height` 为高度。这个文件也可以用 `ebb` 程序生成。`ebb` 是 `dvipdfm` 带的一个工具，专门用来生成图形文件的 `BoundingBox`。MiK_TE_X 和 fp_TE_X 中都已带有这个程序。然后编译下面的例子：

```
\documentclass{article}
\usepackage{graphicx}
\DeclareGraphicsRule{.jpg}{eps}{.bb}{}
\DeclareGraphicsRule{.png}{eps}{.bb}{}
\begin{document}
\includegraphics{ctex-j.jpg}
\includegraphics{ctex-p.png}
\end{document}
```

在 `\includegraphics` 命令中可以使用各种参数，和 EPS 图形一样。编译后使用 `dvipdfm` 将 DVI 转换为 PDF 文件可以看到图形已被正确插入。用最新的 MiK_TE_X 中的 `dvips` 转换为 PostScript 文件或者用 DVI 预览工具 `yap 0.99e` 也可以看到图形，不过图形的大小和颜色都不正确。而 fp_TE_X 的 DVI 预览工具 `windvi` 就完全看不见图形。

54 DVI 里不能显示 EPS 图片？

这是正常现象，因为 EPS 图形并没有真正被加到 DVI 文件中，它是在从 DVI 到 PostScript 转换时才被加到生成的 PostScript 文件中的。因此，所有用到的 EPS 文件必须和 DVI 文件在一起。大多数旧版本的 DVI 浏览器不支持显示 EPS 图形。这时，DVI 浏览器一般会将 EPS 图形的 `BoundingBox` 用一方框显示出来，以方便使用者对图形进行定位。目前版本的一些 T_EX 软件如 MiK_TE_X、fp_TE_X 和 te_TE_X 等所带的 DVI 浏览器 (`Yap`, `Windvi`, `Xdvi`) 可以借助于 `Ghostscript` 来显示 EPS 图形。即使是这样，它们对 EPS 的支持仍然是有限的，不支持 `psfrag` 之类的 PostScript 特殊处理效果，而且经常会出现不正常的显示。此时并不是你的 DVI 文件或者 DVI 浏览器损坏了，而是要将 DVI 文件转换成 PostScript 文件或者 PDF 文件进行察看和打印，才能得到最终的真实效果。

H.2 表格

55 表格中使用 `p` 参数的列如何改变对齐方式？

表格的列参数中的 `p` 参数可以指定列的宽度，但是由于无法和其他控制对齐方式的参数同时作用在一个列上，因此需要使用额外的方法来控制这些列的对齐方式。常用的控制对齐方式命令有：`\centering`（居中）、`\flushleft`（左对齐）和 `\flushright`（右对齐）等。

真正的麻烦并不在这里，而在于换行命令。当 `p` 参数被用在最后一列时，如果在这一列中使用上面的这些对齐命令，例如

```
... & \centering ... \\\
```

就会遇到

```
... misplaced \noalign ...
```

或者类似的编译错误。问题的原因是换行命令 `\\` 在不同的环境下有着不同的定义。在 `tabular` 环境中，这个命令被重新定义以适合于控制表格的显示，而 `\centering`、`\flushleft` 和 `\flushright` 命令把它修改成了另一个不兼容的定义。因为表格的每一个格子都是通过一个盒子（box）来实现的，格子内的定义在分隔符 `&` 或者 `\\` 之后都不再有效，因此上面的这个问题只会出现在每一行的最后一列上。

L^AT_EX 2.09 的解决方法是保存并在稍后恢复 `\\` 的定义：

```
\def\PBS#1{\let\temp=\\%
#1%
\let\\=\temp}
```

用法是

```
... & \PBS\centering ... \\\
```

这种技巧在使用 `array` 宏包提供的格式化列参数方法中也可以使用

```
\begin{tabular}{...>{\PBS\centering}p{50mm}}
...
\end{tabular}
```

而 L^AT_EX 2_ε 则通过公开 `\\` 在表格中的内部定义来解决这个问题。这个定义被保存在命令

```
\tabularnewline
```

中。于是你可以使用

```
... & \centering ... \tabularnewline
```

56 如何生成固定宽度的表格？

在 L^AT_EX 中生成固定宽度的表格有两种方法，一种是通过扩展列之间的空隙，另一种是扩展某些指定的列。

第一种方法通过 `tabular*` 环境来实现。在 `tabular*` 环境的第一个参数中指定表格的总宽度，然后在表格的列格式参数中使用 `\extracolsep` 命令来增加列之间的空隙。

```
\begin{tabular*}{5cm}{@{\extracolsep{\fill}}l11r}
\hline
1 & 2 & 3 & 4 \\\
21 & 22 & 23 & 24 \\\
\hline
\end{tabular*}
```

1	2	3	4
21	22	23	24

这种方法的一个缺点是列之间如果加入分隔符，则列之间的空隙是放在下一列的左边而不是在两列之间平均分配。我们来看上面的例子加入分隔符后的样子

```
\begin{tabular*}{5cm}
  @{\extracolsep{\fill}}|l|l|l|r|}
\hline
  1 & 2 & 3 & 4 \\
  21 & 22 & 23 & 24 \\
\hline
\end{tabular*}
```

1	2	3	4
21	22	23	24

另一种方法可以通过使用 `tabular` 环境的列格式参数之一 `p`(宽度) 来指定相应列的固定宽度。这种方法需要用户手工计算分配每个列的具体宽度。另外, 由于 `p` 参数不能和其他控制对齐方式的参数同时作用在一个列上, 因此需要使用额外的方法来控制这些列的对齐方式 (参见 [问题 55](#))。

`tabularx` 宏包提供了一个自动计算分配列宽度的表格环境 `tabularx`。这个环境的使用方法和 `tabular*` 相同。不同之处在于它增加了一个 `X` 参数。所有使用 `X` 参数的列会平均分配多余的空间, 然后转换为 `p` 参数来指定列的宽度。该宏包也提供了方法让你可以指定如何转换或者转换为其他的列参数。该宏包需要 `array` 宏包的支持。

`ltxtable` 宏包则结合了 `longtable` 和 `tabularx` 宏包的特点。但是它的用法有点复杂, 请仔细阅读它的文档。

H.3 浮动对象

57 浮动对象和上下文之间的距离太大了, 怎么办?

经常有用户抱怨浮动对象 `figure` 或者 `table` 和上下文之间的距离太大了, 以至他们不得不手工调整这个距离。但是大多数人没有意识到是他们自己使用的代码引入了这些额外的空间。

问题出在 `center` 环境 (以及类似的 `flushleft` 和 `flushright` 环境) 上面, 这些环境都是基于 \LaTeX 的 `list` 定义的。 `list` 类型的环境都会在它们自己和上下文之间加入间隔, 这个间隔的大小通过许多参数来决定。因此, 如果我们是用下面的代码:

```
\begin{figure}
  \begin{center}
    \includegraphics{...}
    \caption{...}
  \end{center}
\end{figure}
```

或者更糟糕的

```
\begin{figure}
  \begin{center}
    \includegraphics{...}
  \end{center}
  \caption{...}
\end{figure}
```

那么就会多出一些无法预料的空白。

解决的方法是使用一些基本命令, 而不是使用它们经过 `list` 类型环境包装后的格式。例如

```
\begin{figure}
  \centering
  \includegraphics{...}
  \caption{...}
\end{figure}
```

这个方法适合于所有的 \LaTeX 宏包, 除了一些过时的 \LaTeX 2_{ϵ} 之前的宏包, 如 `psfig` 和 `epsf`。

58 分栏文档中如何使用单栏或者通栏浮动对象？

如果你使用的是 L^AT_EX 标准文档类的 twocolumn 模式，那么使用 `figure` 和 `table` 环境可以产生单列的浮动图形或者表格，而使用 `figure*` 和 `table*` 环境则可以产生通栏的浮动图形或者表格。

如果你使用的是 `multicol` 宏包提供的 `multicols` 环境，那么就有一小问题了。`figure` 和 `table` 在这种方式下无法正常工作，会导致图形或者表格丢失。但是，`figure*` 和 `table*` 仍然可以正常工作，产生通栏的浮动图形或者表格，就像在标准文档类的 twocolumn 模式下一样。

要在 `multicols` 环境中使用单栏的图形或者表格，可以使用 `float` 宏包提供的 `[H]` 位置选项。例如

```
\begin{figure}[H]
...
\caption{...}
\end{figure}
```

但是要注意的是，这种方式产生的对象是不能“浮动”的，也就是说位置是固定的。因此，有时候你必须仔细调整它的位置，以免造成分栏的底部参差不齐。

59 分栏文档中为何不能设定通栏浮动对象的位置？

如果你在在分栏的文档中设定一些通栏的浮动对象（参见 问题 58）的位置选项，你会发现 `[htbp]` 四个选项中的 `h` 和 `b` 选项是不起作用的。也就是说，通栏的浮动对象只能在页面的顶部（`t` 选项）或者单独的页面上（`p` 选项）。以上是对标准文档类的 twocolumn 模式而言。如果使用 `multicol` 宏包的 `multicols` 环境，`b` 选项是有效的，但 `h` 选项也是不起作用。

对于在标准文档类的 twocolumn 模式中需要把通栏浮动对象放置在页面底部，也就是让 `b` 选项有效，可以使用宏包 `stfloats`。

而要让 `h` 选项有效，目前还没有好的办法。`midfloat` 宏包提供的 `strip` 环境可以作为一种近似的解决方案，但效果并不是很好。

I 参考文献

60 如何将文献引用作为上标出现？

将文献引用作为上标出现可以使用 `overcite` 宏包

```
\usepackage{overcite}
```

或者使用 `natbib` 宏包的 `super` 参数

```
\usepackage[super]{natbib}
```

但是，这两种方法产生的上标引用都没有括号。要增加括号，可以修改这两个宏包中的相应定义。更简单的一个方法是使用如下命令

```
\makeatletter
\def\@cite#1#2{\textsuperscript{[#1\if@tempswa , #2\fi]}}
\makeatother
```

而不需使用任何宏包。

61 如何排序并压缩连续的引用，产生如 [2,4–8] 形式的引用？

如果你在 \LaTeX 中使用 $\text{\cite{a,b,c,d,e,f}}$ 则 \LaTeX 不会对条目进行排序，因此，可能会产生象 [6,2,5,8,4,7] 这样很难看的排版效果。大多数人都希望对引用的条目进行排序，并且对连续的条目使用压缩的表示方式，即用 [2,4–8] 表示上面的例子。使用 `cite` 宏包可以解决这个问题。另一种方法是使用宏包 `natbib` 并指定选项 `numbers` 和 `sort&compress`，可以得到一样的效果。如果使用 \BIBTeX 则必须使用 `natbib` 所带的数字式参考文献风格（`plainnat.bst` 和 `unsrtnat.bst`）。

在使用 `hyperref` 宏包来生成超链接的时候会有点问题。`cite` 宏包会完全失效，产生没有排序和压缩的引用。而 `natbib` 则可以得到排序但没有压缩的引用效果。如果你想得到排序而且压缩的效果，可以在加入 `natbib` 宏包后再使用宏包 `hypernat`。

62 如何定制自己的参考文献样式？

用户可以自己编写参考文献的样式文件（`.bst` 文件）。但并不是所有的用户都能迅速掌握用于编写参考文献样式文件的语言。在这种情况下，修改现有的样式文件比较适合对该语言不熟悉的普通用户。如果你需要的样式和现有样式的差别不是太大的话，你可以考虑使用 `custom-bib` 工具。这个工具包含一个 `makebst.tex` 文件。在命令行上输入命令

```
latex makebst
```

在回答一系列问题之后，一个按照你的要求定制的参考文件样式文件就诞生了。

63 如何才能把参考文献条目放在正文中？

一些学术期刊和出版社要求把参考文献条目放在正文中，而另一些可能要求放在脚注里。可以帮助你吧参考文献条目放入正文中的宏包有

- 宏包 `bibentry`，该宏包对你使用的 `.bst` 文件产生的参考文献条目格式有一些限制，但除此之外没有要求使用特定的样式文件。
- 宏包 `inlinebib`，该宏包要求必须使用它自带的 `inlinebib.bst` 样式文件。
- 宏包 `jurabib`，该宏包也要求使用它自带的四个样式文件之一。

而用于把参考文献条目放入脚注中的宏包有

- 宏包 `footbib`
- 宏包 `jurabib`

64 能否生成反向引用（从参考文献指向引用的正文）？

可以。有两个宏包可以使用，一个是 `citeref`，另一个是 `backref` 宏包。`citeref` 的代码似乎相对简单些（因此可能更稳定些），但是不能配合 `hyperref` 生成超链接。而 `backref` 是 `hyperref` 宏包附属的。`backref` 有四个选项可以选择，缺省是 `pageref`：

- `ref` 选项得到引用所在的节编号
- `pageref` 选项得到引用所在的页码
- `hyperref` 选项是超链接形式的 `ref`

- hyperpageref 选项是超链接形式的 pageref

配合 hyperref 宏包使用时可以通过指定 hyperref 的 backref 或者 pagebackref 选项来自动装入 backref 宏包。

两个宏包都只能产生没有压缩的引用（如 [5,6,7] 不会被输出成 [5–6]），同时对于对该参考文献多次引用的页面，也不会重复出现。

65 如何让每章或者每节都有单独的参考文献？

使用 chapterbib 宏包可以让你在一个文档里产生多个参考文献。不过并不像它的名字所表示的那样，这些参考文献并不是和章或者节联系在一起，而是和 \include 命令包含的子文件联系在一起。也就是说，每一个通过 \include 命令包含的子文件可以拥有一个单独的参考文献，

bibunits 宏包则允许用户使用 \bibliographyunit 命令选择让每章还是每节拥有一个单独的参考文献。也可以通过它提供的 bibunit 环境和 \putbib 命令使得任意的一段文档内容拥有一个单独的参考文献。bibunits 必须配合 BibTeX 使用。详细的用法请参考宏包使用说明。

66 如何产生多个参考文献列表？

如果你想让文档的某些部分（如章、节）拥有单独的参考文献，请参考问题 65。这里讨论的是如何将文档的参考文献分割为多个部分，有三种方法可以选择，都必须配合 BibTeX 使用。

multibbl 宏包通过对标准命令的扩展来支持多个参考文献并存。用户可以用 \newbibliography 来定义一个新的参考文献，该命令的参数将会用来生成单独的 .aux 文件，同时用在其他的相关命令中使得 LaTeX 知道使用哪个参考文献。而标准的参考文献命令和引用命令都被修改，增加额外的参数来区分不同的参考文献。其中 \bibliography 命令还增加了用于修改参考文献的标题的参数。该宏包没有包含在 MiKTeX 中，用户可以从 CTAN 下载并安装。

而宏包 multibib 则通过 \newcites 命令让用户定义新的参考文献分类以及分类的标题，而每定义一个新的分类都会产生这个分类专用的参考文献命令。例如，当用户定义了一个新的分类 sec，就会产生四个相应的命令：\bibliographysec、\bibliographystylesec、\citesec 和 \nocitesec。使用这些命令，你就可以把来自一个或者多个参考文献库（.bib 文件）的引用放入任意一个参考文献分类中。

与上面的两个宏包不同，bibtopic 宏包采用另外一种解决方案。用户把参考文献按照自己的分类放入不同的参考文献库（.bib 文件）中，然后通过宏包提供的 btSect 环境指定对应的参考文献库来生成参考文献列表。一个例子如下：

```
\begin{btSect}{english} % 使用 english.bib
\section{References (English)}
\btPrintCited
\end{btSect}
\begin{btSect}{chinese} % 使用 chinese.bib
\section{参考文献 (中文)}
\btPrintCited
\end{btSect}
\begin{btSect}{chinese} % 使用 chinese.bib
\section{未引用的参考文献 (中文)}
\btPrintNotCited
\end{btSect}
```

J 其它专题

J.1 脚注

67 如何在表格中使用脚注？

在标准的 L^AT_EX 表格环境中使用 `\footnote` 会使得脚注内容被吞掉，而不是出现在页面底部。因此脚注无法在表格中使用。如果你用的是浮动表格环境 `table`，那么你必须将表格放到一个 `minipage` 环境中，这样所有的脚注会出现在表格的下方，例如

```
This is text ...

\begin{minipage}{\textwidth}
\begin{tabular}{|c|c|c|}
\hline
aaa\footnote{test} & bbb & ccc \\
\hline
\end{tabular}
\end{minipage}

This is another text ...
```

This is text ...		
aaa ^a	bbb	ccc
— ^a test		
This is another text ...		

另外一个办法是使用宏包 `threeparttable` 中定义的表格。

如果你没有使用浮动环境而只是用了 `tabular` 环境，那么可以使用下面几种方法对表格进行注释。

1. 使用

```
\footnotemark[编号]
```

命令在需要的位置放置脚注的标记，然后在结束 `tabular` 环境以后使用命令

```
\footnotetext[编号]{文字}
```

来加上脚注的内容。这是 Lamport 的书 [6] 中给出的方法。但是如果要做大量的脚注，这个方法就显得有点笨拙了。

2. 使用 L^AT_EX 2_ε 的标准宏包 `tabularx` 或者 `longtable`。这会比用标准的 `tabular` 环境麻烦一点，但是可以正常的使用脚注。

3. 使用 `footnote` 宏包，把 `tabular` 环境放入一个 `savenotes` 环境中，或者在导言区使用

```
\makesavenoteenv{tabular}
```

这样表格中的脚注就会被正确处理。

4. 把表格放在 `minipage` 环境中，这种方法使得所有的脚注都出现在表格的下方（见上面的例子），比较难看。

5. 使用 `threeparttable` 宏包提供的表格。但该宏包是设计为浮动的表格使用的，用在非浮动的表格时会显得有些奇怪。

6. 使用 `mdwtab` 宏包提供的表格，也可以正确处理脚注，还有一些独特功能。但会使得某些和表格有关的宏包无法正常工作。

68 如何在章节标题中使用脚注？

`\footnote` 是一个脆弱的 (fragile) L^AT_EX 命令，这使得当它出现在 `\section` 这类命令的参数中时无法正常工作。在 `\footnote` 之前使用 `\protect` 命令来保护它可以使得 L^AT_EX 正常编译，但是会有副作用：这些脚注标记会出现在目录上，甚至在页眉上，这会进一步带来意想不到的结果！

你无法控制单独这些标记是否只出现在目录或者只出现在页眉上，但可以通过以下的方法使得脚注不出现在目录和页眉上。

- 使用章节标题命令的可选参数，这样只有可选参数中的内容才会出现在目录和页眉中。例如

```
\section[(标题)]{(标题)\footnote{脚注}}
```

- 使用 `footmisc` 宏包并加上 `stable` 参数。这样就可以避免脚注出现在目录和页眉上。

69 如何在图表的标题中使用脚注？

在命令 `\caption` 的参数中使用脚注相当麻烦，可以说是问题 67 和问题 68 的混合体，此外还有它自己的问题。所以决定在标题中使用脚注前最好认真考虑一下是否真的有必要这么做。你必须用可选参数或者其他方法来使得文件能够被 L^AT_EX 处理而且脚注不会出现在目录中，同时也要把浮动的对象放入一个 `minipage` 环境以使得脚注能够被显示出来。

同时你要小心你的脚注是否被打印了两次。假如你使用下面的语句

```
\begin{figure}
  \begin{minipage}{\textwidth}
    ...
    \caption[目录中的标题]{真正的标题\footnote{脚注}}
  \end{minipage}
\end{figure}
```

而其中“真正的标题”足够长（具体的说，需要两行甚至更多行来排版）的时候，L^AT_EX 的结果中就会出现两个一样的脚注。这个讨厌的现象可以通过使用不带 `caption2` 参数的 `ccaption` 宏包来避免。

70 如何重复使用相同内容的脚注？

如果具有相同内容的脚注在文档中多次出现在不同地方，简单地重复输入脚注内容不是一个好办法。我们可以通过半自动的方法来避免重复输入，或者对需要重复的脚注加上标签然后在其他地方加以引用。但是没有完全自动的解决方法（自动发现重复的脚注并加以处理）。

如果你的整个文档只有一个脚注，而且需要重复使用，那么最简单的方法就是使用 `\footnotemark` 命令的可选参数，例如

```
... \footnote{需要重复使用的脚注内容}
...
... \footnotemark[1]
```

可以这么做的原因就是因为我们知道只有一个编号为 1 的脚注。如果脚注不止一个，但是顺序和位置是固定的，那么可以使用类似的技巧，只需重复使用 L^AT_EX 给相应的脚注分配的编号即可。但是大多数时候这么做会比较麻烦，因为文档内容的改变经常会导致脚注之间的关系改变。这时候，使用标签是一个更好的办法。

一个手工标签的方法是使用计数器来记住脚注的编号：

```
\newcounter{fnnumber}
...
... \footnote{需要重复使用的脚注内容}%
\setcounter{fnnumber}{\thefootnote}%
...
... \footnotemark[\thefnnumber]
```

经常这么做确实有点繁琐。L^AT_EX 的标签引用机制可以帮助我们简化工作

```
... \footnote{需要重复使用的脚注内容\label{fn:repeat}}
...
... \footnotemark[\ref{fn:repeat}]
```

但是又有另外一个问题，L^AT_EX 会在第一次运行时报告 `\footnotemark` 命令的参数为空的错误，这是因为 `\ref` 命令无法确定标签的内容而返回了一个空值造成的。这个错误会在第二次运行 L^AT_EX 命令后消失。footmisc 宏包和 memoir 文档类中定义了一个 `\footref` 命令来解决这个问题。这个命令还有一个额外的好处就是即使脚注使用了非数字的标记，它也能正确工作。使用 `\footref` 命令后我们的例子变成了

```
... \footnote{需要重复使用的脚注内容\label{fn:repeat}}
...
... \footref{fn:repeat}
```

这可能是最简洁的解决方法了。需要注意的是在以上的例子里 `\label` 命令必须放在 `\footnote` 的参数中。

fixfoot 宏包提供了另外一种解决机制：使用 `\DeclareFixedFootnote` 命令预先定义好你需要重复使用的脚注，一般都放在导言区中，然后每次只需简单地使用定义好的命令就可以了。

```
\DeclareFixedFootnote{\rep}{需要重复使用的脚注内容}
...
... \rep{}
... \rep{}
```

这个宏包会保证每个重复的脚注在每一页最多出现一次。但是通常你需要运行多次 L^AT_EX 来除去多余的脚注。

J.2 目录

71 怎样象第几章一样，将参考文献也加到目录？

在参考文献部分之前加入

```
\clearpage %双面打印 (openright) 用 \cleardoublepage
\addcontentsline{toc}{chapter}{参考文献}
```

语句即可。类似的，你也可以手工加入其它目录项。

72 手工加入的目录项页码错误，生成的书签链接也不对，怎么回事？

这是因为 `\addcontentsline` 命令后面的命令换页造成的页码不一致。需要在 `\addcontentsline` 命令之前加上

```
\clearpage
```

如果是 book 类或者用了 openright 参数的 report 类文档，新的章总是从奇数页开始，则要把命令改成

```
\cleardoublepage
```

如果使用了 hyperref 宏包来生成 PDF 文件的书签链接，则还需要在上面的命令后再加上

```
\phantomsection
```

J.3 特殊符号

73 如何输出 ^ 和 ~ 符号？

^ 和 ~ 符号都是 L^AT_EX 的特殊符号，可以在 verbatim 环境得到符号本身，例如

```
\verb|^| and \verb|^| is special characters.
```

```
^ and ~ is special characters.
```

还可以使用命令

```
\textasciicircum
```

得到 ^ 符号，用

```
\textasciitilde
```

得到 ~ 符号。

74 如何显示可见的空格符号 _ ？

可以使用

```
\textvisiblespace
```

命令，例如

```
Hello\textvisiblespace world\textvisiblespace !
```

```
Hello_world_!
```

75 如何显示带圈的符号？

可以使用

```
\textcircled{<letter>}
```

为了产生比较美观的符号，可能要对其中的字体大小做一定的调整。

```
\textcircled{\scriptsize 1}
\textcircled{\scriptsize 2}
\textcircled{3}
\textcircled{4}
\textcircled{\scriptsize a}
\textcircled{\scriptsize b}
\textcircled{c}
\textcircled{d}
```

```
① ② ③ ④ a b c d
```

实际上版权符号 © 就是这么产生的，只是为了方便用户专门预定义了命令 \copyright。你也可以为自己常用的一些符号定义一个简短的命令。

K 高级技术

76 `\makeatletter` 和 `\makeatother` 是干什么用的？

一些 \LaTeX 内部命令含有 `@` 字符，如 `\@addtoreset`，如果需要在文档中使用这些内部命令，就需要借助于另两个命令 `\makeatletter` 和 `\makeatother`。

下面的例子可以实现公式编号与节号的关联：

```
\documentclass{article}
...
\makeatletter % @ is now a normal "letter" for TeX
\renewcommand\theequation{\thesection.\arabic{equation}}
\@addtoreset{equation}{section}
\makeatother % @ is restored as a "non-letter" character for TeX
\begin{document}
...
```

77 如何查询 \LaTeX 命令和参数的缺省定义？

首先你可以直接搜索相应的 \LaTeX 宏包以及同一目录下的其他文件。如果你有宏包的源文件 (`.dtx`) 的话，搜索源文件会更方便些。而且源文件中会有详细的说明。对于这些源文件，可以直接用 \LaTeX 编译，产生说明文档。 \LaTeX 宏包的源文件一般放在

```
texmf\source\latex\
```

目录下。例如 \LaTeX 标准文档类的源文件就是

```
texmf\source\latex\base\classes.dtx
```

如果这些文件里都找不到定义，则有可能在 \LaTeX 内核中定义。你可以搜索

```
texmf\source\latex\base\
```

目录下的其他 `.dtx` 文件，所有的 \LaTeX 内核定义都在那里了。如果还没有的话，那就可能是 \TeX 的基本命令或者定义了，查询 Donald E. Knuth 的 *The \TeX book* ^[7]，[CTEX FTP](#) 上可以下载该书的源文件。

L 辅助工具

78 WinEdt 和 GSview 经常出现过期和注册提示，如何解决？

WinEdt 和 GSview 是共享软件，需要注册才能使用全部功能，如果你不能忍受它的提示，那就只有花钱注册了。

79 如何去掉 WinEdt 的自动断行和自动续行？

点击 WinEdt 状态栏上的 Wrap 使其变为灰色可以完全禁止 WinEdt 的自动断行和自动续行。在行首加入空格或者 Tab 符可以防止自动续行（将本行并入上一行）。

在 WinEdt 的菜单 Options 下选择 Preferences，在弹出的对话框中的 Editor 页的右下角可以设置 Right Margin 来修改 WinEdt 判断自动断行的字符数。

80 如何让 WinEdt 显示行号?

点击 WinEdt 状态栏上的总行数（左数第四格）可以打开和关闭行号的显示。

New
v0.4

81 Yap 里如何预览带有颜色的 EPS 图片?

打开 Yap 的菜单 Views 下的 Options, 在弹出的对话框中选择 PostScript, 然后在 Quality 的下拉框中选择 16 million colors.

M 错误处理

82 Error: pdflatex (file simsun.ttf): can't find table 'maxp'

这个错误是因为中文 Windows 2000 及以后的操作系统的宋体字库使用了新的 TrueType 格式的文件 simsun.ttc, 而 pdflatex 不支持该文件格式。解决方法可以选以下四种之一:

New
v0.4

1. 不使用 pdflatex 而改用其他方法生成 PDF 文件
2. 不使用宋体改用其他字体
3. 不使用 TrueType 字库而改用 Type1 字库（安装 C_TE_X-Fonts）
4. 从中文 Windows 98 下拷贝一个 simsun.ttf 到 Windows 安装目录下的 fonts 目录或者 C_TE_X 的安装目录下的 locatexmf 的相应子目录下

N 帮助文档

83 能否推荐一些比较好的中文资料?

目前 T_EX 和 L^AT_EX 的中文资料还不是很多, 特别是相对大量的英文资料而言。如果你刚开始学习 L^AT_EX, 我们推荐你使用《lshort 中文版》^[1]。这是一份很好的入门学习教材。它的最新版本可以在 [C_TE_X 网站](#) 免费下载。

如果你需要一本详细的 L^AT_EX 命令手册, 我们推荐使用邓建松等人翻译整理的《L^AT_EX 2_ε 科技排版指南》一书^[2]。这本书可以说是目前国内正式出版的 L^AT_EX 书籍中最值得购买的一本。它的一份电子版初稿也可以在 [C_TE_X 网站](#) 免费下载。

另外一本值得推荐的书是《L^AT_EX 2_ε 插图指南》^[3]。它是由 C_TE_X 的王磊在 Keith Reckdahl 编写的 Using Import graphics in L^AT_EX 2_ε (Ver. 2, Dec. 1997) 一书的基础上翻译整理的。该书目前尚未正式出版, 其电子版可以在 [C_TE_X 网站](#) 免费下载。

84 能否推荐一些比较好的英文资料?

如果你刚开始学习 L^AT_EX, 我们推荐使用 lshort (The Not So Short Introduction to L^AT_EX 2_ε)^[5]。与其他的书相比, 这份小册子相对比较简短, 但覆盖了所有常用的 L^AT_EX 2_ε 命令, 而且附有大量的实例。lshort 一直在不断更新, 而且有十几个语言的翻译版本。你可以把英文版和中文版^[1] 对照着看, 这样会对一些专业名词更加容易理解。

如果你打算详细全面的学习 L^AT_EX, 最经典的教材莫过于 L^AT_EX 的作者 Leslie Lamport 的著作 L^AT_EX: A Document Preparation System^[6] 了。另一本更加详细的介绍 L^AT_EX 及其宏包的书籍就是 The L^AT_EX Companion^[8], 这本 500 多页的书中包含了超过 150 个宏包的描述。这两本书没有在国内出版, 但是你可以在一些图书馆中借到。The L^AT_EX Companion 的第八章是关于数学公式排版的介绍, 有一份电子版可以在 CTAN 上下载, 路径是

```
CTAN:/info/companion-rev
```

或者用 `ch8.pdf` 在 CTAN 上进行搜索 (参见 问题 14)。

L^AT_EX 系统自带了许多文档,除了各个宏包的说明以外,还有一些是关于 L^AT_EX 系统的说明。这些文档都放在

```
texmf\doc\latex\base
```

目录下。其中, L^AT_EX 2_ε for Authors ^[9] 是介绍如何用 L^AT_EX 2_ε 来排版的, L^AT_EX 2_ε for Class and Package Writers ^[10] 则介绍了如何利用 L^AT_EX 2_ε 来编写宏包,这两份文档侧重点都在于描述 L^AT_EX 2_ε 的新特性。如果你是 L^AT_EX 2.09 的用户,这两份文档会有很大的帮助。L^AT_EX 2_ε Font Selection ^[11] 则介绍了 L^AT_EX 2_ε 的字体选择方案,它会帮助你了解 L^AT_EX 2_ε 是如何定义、使用字体的。

如果你对 L^AT_EX 底层的東西开始感兴趣,或者你开始想自己修改或者编写一些 L^AT_EX 宏包,那么 T_EX 的作者 Donald E. Knuth 的巨著 *The T_EXbook* 实在是必不可少。*The T_EXbook* 同时也是一部漂亮的排版杰作,而且还能让我们接触到一代宗师 Knuth 卓越不凡的思想与理念。

85 有没有关于 L^AT_EX 宏包的详细说明?

T_EX 系统安装时候带的 L^AT_EX 宏包,都可以在 T_EX 的安装目录下的

```
texmf\doc\latex
```

目录下找到帮助文件和例子。

CT_EX 的 [在线文档](#) 提供了一份关于常见的 L^AT_EX 宏包的列表和说明。另外, Graham Williams 维护的 *The TeX Catalogue* ^[12] 是一份全面的 T_EX 和 L^AT_EX 宏包分类目录,这份文档可以在 CTAN 的镜像服务器上下载 (参见 问题 11 和 问题 14)。

86 我的问题没找到答案,或者有答案但是解决不了我的问题,怎么办?

欢迎到 [CT_EX 论坛](#) 上提出你的问题,也欢迎指出这份文档中不正确或者不准确的地方。提问的时候请提供尽可能详细的信息,例如使用的 T_EX 系统、操作系统、使用的文档类和宏包命令、简单而又可以重复错误现象的例子等等,而不是仅仅说“为什么我的 DVI 中看不到中文”这样一句话,这种问题没有人能够回答你。

参考文献

- [1] CT_EX 翻译小组。 **lshort 中文版 3.20**, 2003。
- [2] 邓建松, 彭冉冉, 陈长松。 **L^AT_EX 2_ε 科技排版指南**, 科学出版社, 书号: 7-03-009239-2/TP.1516, 2001。
- [3] 王磊。 **L^AT_EX 2_ε 插图指南**, 2000。
- [4] 张林波。 **关于新版 CCT 的说明**, 2003。
- [5] Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl. **The Not So Short Introduction to L^AT_EX 2_ε 4.12**, 2003。
- [6] Leslie Lamport. **L^AT_EX: A Document Preparation System**. Addison-Wesley, Reading, Massachusetts, second edition, 1994, ISBN 0-201-52983-1。

-
- [7] Donald E. Knuth. **The T_EXbook**, Volume A of *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts, second edition, 1984, ISBN 0-201-13448-9.
- [8] Michel Goossens, Frank Mittelbach and Alexander Samarin. **The L^AT_EX Companion**. Addison-Wesley, Reading, Massachusetts, 1994, ISBN 0-201-54199-8.
- [9] L^AT_EX3 Project Team. **L^AT_EX 2_ε for Authors**. L^AT_EX 2_ε 随机文档 `usrguide.tex`
- [10] L^AT_EX3 Project Team. **L^AT_EX 2_ε for Class and Package Writers**. L^AT_EX 2_ε 随机文档 `clsguide.tex`
- [11] L^AT_EX3 Project Team. **L^AT_EX 2_ε Font Selection**. L^AT_EX 2_ε 随机文档 `fntguide.tex`
- [12] Graham Williams. **The T_EX Catalogue**. 一份全面的 T_EX 和 L^AT_EX 宏包分类目录, 在线访问 CTAN:/help/Catalogue/catalogue.html
- [13] Robin Fairbairns. **The UK T_EX FAQ**. 英国 T_EX 用户协会 FAQ.

索引

- `\\`, 23
- abstract, 16
- `\addcontentsline`, 30
- amsmath, 19
- amsmath, 6, 18–20
- array, 23, 24
- `\author`, 16
- backref, 26, 27
- balance, 15, 16
- `\balance`, 16
- `\baselinestretch`, 14
- bibentry, 26
- `\bibliographyunit`, 27
- bibtopic, 27
- bibunit, 27
- bibunits, 27
- bm, 19
- `\bm`, 19
- btSect, 27
- `\caption`, 21, 29
- ccaption, 29
- center, 24
- `\centering`, 22, 23
- chapterbib, 27
- cite, 26
- `\cite`, 26
- citeref, 26
- CJK, 12–14
- CJK, 12, 13
- CJK*, 12, 13
- `\CJKglue`, 12
- `\CJKindent`, 12
- `\CJKnospace`, 13
- `\CJKspace`, 13
- `\CJKtilde`, 13
- `\cleardoublepage`, 31
- `\clearpage`, 15, 30
- comment, 15
- `\copyright`, 31
- `\DeclareFixedFootnote`, 30
- `\documentclass`, 7, 11, 15
- `\documentstyle`, 6, 7
- DVI, 22
- dvipdfmx, 14
- ebb, 22
- EPS, 22, 33
- epsf, 24
- `\extracolsep`, 23
- `\fi`, 15
- figure, 24, 25
- figure*, 25
- fixfoot, 30
- float, 25
- flushend, 15, 16
- `\flushend`, 15
- `\flushleft`, 22, 23
- flushleft, 24
- `\flushright`, 22, 23
- flushright, 24
- footbib, 26
- footmisc, 29, 30
- footnote, 28
- `\footnote`, 28–30
- `\footnotemark`, 28–30
- `\footnotetext`, 28
- `\footref`, 30
- `\hskip`, 12
- hypernat, 26
- hyperref, 26, 27
- `\iffalse`, 14
- `\include`, 27
- `\includegraphics`, 21, 22
- indentfirst, 14
- inlinebib, 26
- jurabib, 26
- `\label`, 30
- landscape, 17
- latexsym, 7
- `\limits`, 19, 20

- list, 24
- \listoffigures, 21
- longtable, 24, 28
- lscap, 17
- ltxtable, 24

- \makeatletter, 32
- \makeatother, 32
 - makebst, 26
- \makesavenoteenv, 28
- \maketitle, 16
- \mathbb, 19
- mathtime, 17
- mdwtab, 28
- memoir, 30
- midfloat, 25
- minipage, 28, 29
- multibbl, 27
- multibib, 27
- multicol, 15, 16, 25
- multicols, 15, 25

- natbib, 25, 26
- \newbibliography, 27
- \newcites, 27
- \newpage, 15
- \nobalance, 16
- \nolimits, 20
- \numberwithin, 18

- onecolabstract, 16
- overcite, 25

- PDF, 14
- \phantomsection, 31
 - preamble, 10
- \protect, 29
 - psfig, 24
- \putbib, 27

- \raggedend, 15
- \ref, 30
- \renewcommand, 12, 14
 - savenotes, 28
- \saythanks, 16
- \section, 29

- \setcounter, 18
- \standardtilde, 13
- stfloats, 25
- strip, 25
- subequations, 18

- table, 24, 25, 28
- table*, 25
- tabular, 23, 24, 28
- tabular*, 23, 24
- \tabularnewline, 23
- tabularx, 24, 28
- tabularx, 24
- \tag, 19
- \textasciicircum, 31
- \textasciitilde, 31
- \textcircled, 31
- \textvisiblespace, 31
- \thanks, 16
- \theequation, 18
- threeparttable, 28
- times, 17
- \title, 16
 - TrueType, 13, 14
 - Type1, 13, 14
- \usepackage, 7

- verbatim, 15
- verbatim, 31

- WinEdt, 32, 33

- \xleftarrow, 20
- \xrightarrow, 20

- Yap, 33

- 导言区, 10

- 宏包 (Packages)
 - abstract, 16
 - amsmath, 6, 18–20
 - array, 23, 24
 - backref, 26, 27
 - balance, 15, 16
 - bibentry, 26

- bibtopic, 27
 - bibunits, 27
 - bm, 19
 - ccaption, 29
 - chapterbib, 27
 - cite, 26
 - citeref, 26
 - CJK, 12–14
 - epsf, 24
 - fixfoot, 30
 - float, 25
 - flushend, 15, 16
 - footbib, 26
 - footmisc, 29, 30
 - footnote, 28
 - hypernat, 26
 - hyperref, 26, 27
 - indentfirst, 14
 - inlinebib, 26
 - jurabib, 26
 - latexsym, 7
 - longtable, 24, 28
 - lscape, 17
 - ltxtable, 24
 - mathtime, 17
 - mdwtab, 28
 - memoir, 30
 - midfloat, 25
 - multibbl, 27
 - multibib, 27
 - multicol, 15, 16, 25
 - natbib, 25, 26
 - overcite, 25
 - psfig, 24
 - stfloats, 25
 - tabularx, 24, 28
 - threeparttable, 28
 - times, 17
 - verbatim, 15
- 环境 (Environments)
- bibunit, 27
 - btSect, 27
 - center, 24
 - CJK, 12, 13
 - CJK*, 12, 13
 - comment, 15
 - figure, 24, 25
 - figure*, 25
 - flushleft, 24
 - flushright, 24
 - landscape, 17
 - list, 24
 - minipage, 28, 29
 - multicols, 15, 25
 - oncolabstract, 16
 - savenotes, 28
 - strip, 25
 - subequations, 18
 - table, 24, 25, 28
 - table*, 25
 - tabular, 23, 24, 28
 - tabular*, 23, 24
 - tabularx, 24
 - verbatim, 31
- 命令 (Commands)
- \\, 23
 - \addcontentsline, 30
 - \author, 16
 - \balance, 16
 - \baselinestretch, 14
 - \bibliographyunit, 27
 - \bm, 19
 - \caption, 21, 29
 - \centering, 22, 23
 - \cite, 26
 - \CJKglue, 12
 - \CJKindent, 12
 - \CJKnospace, 13
 - \CJKspace, 13
 - \CJKtilde, 13
 - \cleardoublepage, 31
 - \clearpage, 15, 30
 - \copyright, 31
 - \DeclareFixedFootnote, 30
 - \documentclass, 7, 11, 15
 - \documentstyle, 6, 7
 - \extracolsep, 23
 - \fi, 15
 - \flushend, 15
 - \flushleft, 22, 23

`\flushright`, 22, 23
`\footnote`, 28–30
`\footnotemark`, 28–30
`\footnotetext`, 28
`\footref`, 30
`\hskip`, 12
`\iffalse`, 14
`\include`, 27
`\includegraphics`, 21, 22
`\label`, 30
`\limits`, 19, 20
`\listoffigures`, 21
`\makeatletter`, 32
`\makeatother`, 32
`\makesavenoteenv`, 28
`\maketitle`, 16
`\mathbb`, 19
`\newbibliography`, 27
`\newcites`, 27
`\newpage`, 15
`\nobalance`, 16
`\nolimits`, 20
`\numberwithin`, 18
`\phantomsection`, 31
`\protect`, 29
`\putbib`, 27
`\raggedend`, 15
`\ref`, 30
`\renewcommand`, 12, 14
`\saythanks`, 16
`\section`, 29
`\setcounter`, 18
`\standardtilde`, 13
`\tabularnewline`, 23
`\tag`, 19
`\textasciicircum`, 31
`\textasciitilde`, 31
`\textcircled`, 31
`\textvisiblespace`, 31
`\thanks`, 16
`\theequation`, 18
`\title`, 16
`\usepackage`, 7
`\xleftarrow`, 20
`\xrightarrow`, 20

Contributors

吴凌云 (aloft@ctex.org)

Copyright ©2003–2004 C_TE_X and all the Contributors to C_TE_X FAQ. All rights reserved.

C_TE_X 以及 C_TE_X FAQ 的作者拥有版权 ©2003–2004。保留所有权利。

Permission is granted to copy, distribute and/or modify this document under the terms of the [GNU Free Documentation License](#), Version 1.2 or any later version published by the [Free Software Foundation](#); with the Invariant Sections being Contributors, no Front-Cover Texts, and no Back-Cover Texts.

你可以拷贝、发布或者修改这份文档，但必须遵守[自由软件组织](#)颁布的 [GNU 自由文档许可证](#) 1.2 或者以后版本的条款。Invariant Sections 包括 Contributors，没有 Front-Cover Texts 和 Back-Cover Texts。

ChangeList of v0.4

New

17	CT _E X 中文套装软件的 Basic、Pro、Full 版本都有什么区别?	8
18	CT _E X-Fonts、CT _E X-CS、CT _E X-Ext 都是干什么的? 和 CT _E X 中文套装有什么关系?	9
80	如何让 WinEdt 显示行号?	33
82	Error: pdflatex (file simsun.ttf): can't find table 'maxp'	33

Revised

3	这个 FAQ 的目的是什么?	5
20	能否举一个简单的英文排版例子?	9
38	如何生成分栏的文档?	15
39	如何使得双栏文档保持底部对齐?	15
43	如何使用 New Time Roman 字体?	17
52	L ^A T _E X 如何使用图形?	21

(Last updated: September 27, 2004)