

标题页

Ryan* Fan†

2020 年 4 月 28 日

摘要

一般用于紧跟\maketitle 命令之后介绍文档的摘要
中文 L^AT_EX 排版。

1 用 L^AT_EX 排版文字

分段

换行\\

冒号 “please press the ‘x’ key.”

连字符-用来组成复合词,

短破折号—用来连接数字表示范围,

长破折号—用来连接单词

省略号...和...

波浪号~

强调文字, 但是无法换行, `ulem` 宏包解决了这一问题, 它提供的 `uline` 命令能够轻松生成自动换行的下划线。`emph` 命令用来将文字变为斜体以示强调。如果在本身已经用 `emph` 命令强调的文字内部嵌套使用 `emph` 命令, 内部则使用直立体文字。

在合适的位置插入一个不会断行的空格 Fig. 1, Ryan Fan

断行

可以带可选参数 `length`, 用于在换行处向下增加垂直间距
或者 `newline` 命令, 不用带参数

*注脚

†注脚

断页，在在双栏排版中只起到另起一栏的作用

断词 I think this is: supercalifragilisticexpialidocious. And I think
this is: supercalifragilisticexpialidocious.

目录

2 章节

2.1 子章节

2.1.1 子子章节

段落

子段落

标题不带编号

第一部分 分块

3 交叉引用

A reference to this subsection looks like: “see section ?? on page ??.”

4 脚注和边注

“天地玄黄, 宇宙洪荒。日月盈昃, 辰宿列张。”¹
有些情况下 (比如在表格环境、各种盒子内) 使用 footnote 并不能正确生成脚注。我们以分两步进行, 先使用 footnotemark 为脚注计数, 再在合适的位置用 footnotetext 生成脚注。²

边注较窄, 不要写过多文字, 最好设置较小的字号。

5 特殊环境

5.1 列表

有序列表

- 1. An item.
 - (a) A nested item.
 - * A starred item.
- 2. Reference(??).

¹出自《千字文》。

²出自《千字文》。

无序列表

- An item.
 - A nested item.
 - + A ‘plus’ item. + A ‘plus’ item.
 - Another item. –Another item.
- Go back to upper level.

关键字环境

Enumerate Numbered list.

Itemize Non-numbered list.

重定义无序列表的符号

- † First item
 - ‡ Subitem
 - ‡ Subitem
- † Second item

重定义有序列表的符号

- A> First item
- B> Second item

5.2 对齐环境

`center`、`flushleft` 和 `flushright` 环境分别用于生成居中、左对齐和右对齐的文本环境。

Centered text using a **center** environment.

Left-aligned text using a **flushleft** environment.

Right-aligned text using a **flushright** environment.

还可以用以下命令直接改变文字的对齐方式: Centered text paragraph.

Left-aligned text paragraph.

Right-aligned text paragraph.

`center` 等环境会在上下文产生一个额外间距, 而 `\centering` 等命令不产生, 只是改变对齐方式。

比如在浮动体环境 `table` 或 `figure` 内实现居中对齐, 用 `\centering` 命令即可, 没必要再用 `center` 环境。

5.3 引用环境

quote 用于引用较短的文字, 首行不缩进

Francis Bacon says:

Knowledge is power.

quotation 用于引用若干段文字, 首行缩进

《木兰诗》:

万里赴戎机, 关山度若飞。朔气传金柝, 寒光照铁衣。将
军百战死, 壮士十年归。
归来见天子, 天子坐明堂。策勋十二转, 赏赐百千强。

.....

5.4 代码环境

```
#include <iostream>
int main()
{
    std::cout << "Hello, world!"
                << std::endl;

    return 0;
}
```

```
for(int i=0; i<4; i++)
printf("Number_%d\n", i);

```

要排版简短的代码或关键字`\verb delim code delim`

`delim` 标明代码的分界位置, 前后必须一致, 除字母、空格或星号外, 可任意选择使得不与代码本身冲突, 习惯上使用 `|` 符号。

`\LaTeX`

`(a || b) (a_|||_b)`

5.5 表格

5.5.1 列表格

`tabular` 环境使用 `column-spec` 参数指定表格的列数以及每列的格式。

left	center	right	par box with fixed width
L	C	R	P

@ 格式可在单元格前后插入任意的文本, 但同时它也消除了单元格前后额外添加的间距。

1:1	one
11:3	eleven

格式参数重复

one	two	three	four	five	Hello!	Hello!
1	2	3	4	5	L ^A T _E X	L ^A T _E X

辅助格式 `>` 和 `<`, 用于给列格式前后加上修饰命令

*italic** normal

*column** column

辅助格式甚至支持插入 `\centering`

等命令改变 p 列格式的对齐方式, 一

般还要加额外的命令

`\arraybackslash` 以免出错。

`\centering` 等对齐命令会破坏表格环

境里 `\\` 换行命令的定

义, `\arraybackslash` 用来恢复之。如

果不加 `\arraybackslash` 命令, 也可

以用 `\tabularnewline` 命令代替原来

的 `\\` 实现表格换行。

L^AT_EX 本身提供了 `tabular*` 环境用来排版定宽表格, 但是不太方便使用, 比如要用到 @ 格式插入额外命令, 令单元格之间的间距为 `\fill`, 但即使这样仍然有瑕疵:

A	B	C	D
a	b	c	d

`tabularx` 宏包为我们提供了方便的解决方案。它引入了一个 X 列格式, 类似 p 列格式, 不过会根据表格宽度自动计算列宽, 多个 X 列格式平均分配列宽。X 列格式也可以用 `array` 里的辅助格式修饰对齐方式:

A	B	C	D
a	b	c	d

5.5.2 横线

`\cline{i-j}` 用来绘制跨越部分单元格的横线:

三线表由 `booktabs` 宏包支持, 它提供了 `\toprule`、`\midrule` 和

`\bottomrule` 命令用以排版三线表的三条线, 以及和 `\cline` 对应的

`\cmidrule`。除此之外, 最好不要用其它横线以及竖线:

XX

测试