

# 第八讲 细节排版

潘建瑜

华东师范大学数学系

[jypan@math.ecnu.edu.cn](mailto:jypan@math.ecnu.edu.cn)

2014

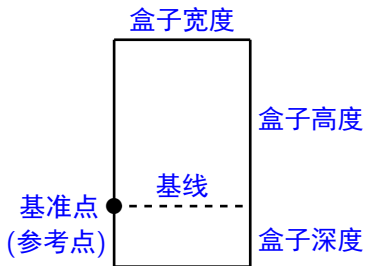
# 本讲内容

---

- ① 盒子
- ② 段落盒子与小页环境
- ③ 引文与抄录
- ④ 表格高级技巧
- ⑤ 罗列
- ⑥ 角注与边注

# 盒子

☞ 在  $\text{\LaTeX}$  中, 一切都是盒子!



- $\text{\LaTeX}$  的排版过程就是在构造盒子和堆砌盒子
  - 当盒子左右排列时, 它们的基线在同一水平线上
  - 当盒子上下排列时, 基准点在同一垂直线上
- 每个盒子有三个尺寸: 宽度 `\width`, 高度 `\height` 和深度 `\depth`
- 总高度 `\totalheight = \height + \depth`
- 盒子无论是否带框, 都占据一个矩形区域

# 盒子

☞ 构造盒子: 用户可以自己构造的盒子有

- LR 盒子 (左右盒子): 内容只能位于同一行 (单行盒子)
- 段落盒子: 由垂直堆叠的行构成, 可包含多行文本 (多行盒子)
- 标尺盒子: 为一个实心矩形, 通常用于画水平线或垂直线

☞ LR 盒子构造命令

`\mbox{单行文本}` → 普通单行盒子

`\fbox{单行文本}` → 带框单行盒子

`\makebox[宽度][位置]{单行文本}` → 指定宽度的普通单行盒子

`\framebox[宽度][位置]{单行文本}` → 指定宽度的带框单行盒子

- **单行文本**: 不能分行, 但可以是一个环境或盒子
- 前两个命令生成的盒子宽度等于文本的实际宽度, 即**自然宽度**
- 后两个命令可指定盒子的宽度和盒内文本的排放位置
  - 排放位置: l (居左), r (居右), s (拉伸), 缺省为居中
  - 后两个命令的选项省略一个时, 只能省略**位置**

# LR 盒子


## 🗨️ 几点说明

- 若指定的宽度小于文本**自然宽度**时, 文本将会超出盒子
- 设置盒子框线与文本之间的距离以及盒子框线的粗细

```
\setlength{\fboxsep}{长度} → 设置框线与文本间距  
\setlength{\fboxrule}{宽度} → 设置框线粗细
```

- 指定盒子宽度时, 可以使用盒子的**自然尺寸** (`\width`)
- 盒子宽度为 0 时可以实现一些特殊的效果
- 例 61box.tex

## 盒子的升降

 盒子的升降: 在竖直方向移动一定的距离

```
\raisebox{升高量}[盒高][盒深]{文本}
```

- 该命令生成一个无框的盒子, 盒子的基线位于当前行的基线上, 但盒子内文本的基线相对于盒子的基线在竖直方向作一定的移位
- 升高量为正 → 将盒子上移  
升高量为负 → 将盒子下移
- 选项盒高和盒深缺省时, 自动调整大小
- 若升高量过大, 文本可能会超出盒子的范围
- 盒高和盒深的值为负时, 被当成 0 处理
- 同一行中盒子的基线位于同一水平线上, 这些盒子组成一个更大的盒子, 大盒子的高度和深度分别为各小盒子高度和深度的最大值
- 高度和深度为 0 的升降盒子对于排版一些特殊的表格或矩阵非常有用。

## 盒子的升降

```
\fbox{正常文本}
\fbox{\raisebox{5pt}{升高 5pt}}\
\fbox{正常文本}
\fbox{\raisebox{-5pt}{降低 5pt}}
```

正常文本	升高 5pt
正常文本	降低 5pt

```
\begin{tabular}{|c|c|c|}\hline
  居中 & hello & world \\ \cline{2-3}
        & Math  & ECNU  \\ \hline
\end{tabular}
```

居中	hello	world
	Math	ECNU

```
\begin{tabular}{|c|c|c|}\hline
  \raisebox{-7pt}[0pt][0pt]{居中}
  & hello & world \\ \cline{2-3}
        & Math  & ECNU  \\ \hline
\end{tabular}
```

居中	hello	world
	Math	ECNU

① 盒子

② 段落盒子与小页环境

③ 引文与抄录

④ 表格高级技巧

⑤ 罗列

⑥ 角注与边注



## 段落盒子与小页环境

👉 段落盒子与小页环境: 将整段文本放入一个盒子的两种方法

👉 段落盒子的构造

```
\parbox[位置]{宽度}{文本}  
\parbox[位置][总高度][内部位置]{宽度}{文本}
```

- 生成一个指定宽度的段落盒子
- 排版时, 盒子被看成一个不可分割的整体
- **位置**用来指定盒子与外部文本的对齐方式:
  - **b** → 盒子底行文本基线与盒外基线对齐
  - **t** → 盒子顶行文本基线与盒外基线对齐
  - 缺省为垂直居中, 即盒子中部与盒外基线对齐
- 可通过选项**总高度**指定盒子的总高度, 缺省为所含内容的**自然高度**
  - 可选项**内部位置**用于指定内部文本的垂直排放方式, 可以为: **t, b, c, s**
  - 可选项**总高度**和**内部位置**应同时使用或同时不用

# 小页环境

---


## 小页环境

```
\begin{minipage} [位置] {宽度}  
  ...  
\end{minipage}
```

```
\begin{minipage} [位置] [总高度] [内部位置] {宽度}  
  ...  
\end{minipage}
```

- 各选项的含义与 `\parbox` 相同

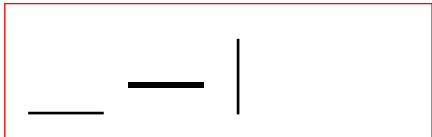
# 标尺盒子

 标尺盒子: 一个实心矩形, 可用于画横线和数线

```
\rule[升高量]{水平宽度}{竖直宽度}
```

- 可选项**升高量**指定标尺盒子底部相对于当前行基线向上移动的距离
- 宽度为 0 的标尺盒子是看不见的, 但可起“支撑”作用

```
\rule{1cm}{1pt}\quad  
\rule[10pt]{1cm}{2pt} \quad  
\rule{1pt}{1cm}
```



## 盒子的重复使用

☞ 盒子的重复使用: 创建一个盒子, 保存后, 可以在不同地方多次使用

- (1) 创建新盒子: `\newsavebox{盒子名}`
- (2) 将文本保存到该盒子中

```
\sbox{\盒子名}{文本}
\savebox{\盒子名}[宽度][位置]{文本}
```

- (3) 使用盒子: `\usebox{盒子名}`
  - 给盒子加框: `\fbox{\usebox{盒子名}}`
  - 排版时,  $\text{\LaTeX}$  只按盒子的尺寸安排所占空间


☞ 如果需要保存大段文本到一个盒子里, 可以使用 `lrbox` 环境

```
\begin{lrbox}{\盒子名}
... ..
\end{lrbox}
```

该环境的作用与 `\sbox` 一样

- ① 盒子
- ② 段落盒子与小页环境
- ③ 引文与抄录
- ④ 表格高级技巧
- ⑤ 罗列
- ⑥ 角注与边注


## 引文与抄录

 引文环境: 另起一段, 并在两边向内缩进一定距离

```
\begin{quotation}
... ..
\end{quotation}
```

```
\begin{quote}
... ..
\end{quote}
```

- 引文可以是一行, 或多行, 甚至几个段落
  - `quote` 中的段落首行不自动缩进, 但增加段间距

 西文诗歌, 韵文环境

```
\begin{verse}
... ..
\end{verse}
```

```
this is normal text normal text
\begin{quotation}
this is quotation
this is quotation
this is quotation
\end{quotation}
```

this is normal text normal text

*this is quotation this is quotation this is quotation*

```
this is normal text normal text
\begin{quote}
this is quote this is quote

this is quote this is quote
\end{quote}
```

this is normal text normal text

*this is quote this is quote*

*this is quote this is quote*

```
this is normal text normal text
\begin{verse}
this is verse this is verse
this is verse
\end{verse}
```

this is normal text normal text

*this is verse this is verse*

*this is verse*

## 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  $\LaTeX$  命令
- 所有的  $\LaTeX$  命令都被作为普通字符串输出
- 西文用打字机字体, 汉字为进入该环境前的字体
- 带星号的环境用  $\_$  显示空格
- 例: 下面的内容就是通过抄录环境输出的

---

```
\begin{equation}
\exp(ix)=\cos(x)+i\sin(x).
\end{equation}
```

---



## 抄录命令:

```
\verb|抄录文本|  
\verb*|抄录文本|
```

- 适用于不超过一行的文本, 即抄录文本不能多于一行
- 带星号的命令用 `_` 显示空格
- 定界符 `|` → 可以使用除星号和空格以外的任何抄录文本中没有出现的符号, 但左右定界符必须一致
- 抄录环境和抄录命令都不能作为其它命令的参数

```
\verb|hello $\frac{1}{2}$ hello|
```

```
hello $\frac{1}{2}$ hello
```

```
\verb*[hello $\frac{1}{2}$ hello]
```

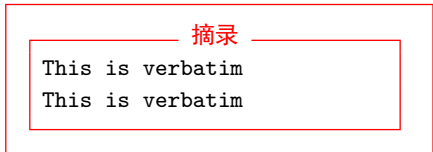
```
hello_ $\frac{1}{2}$ _hello
```

# fancyvrb 宏包

 功能丰富的抄录: fancyvrb 宏包, 提供 Verbatim 环境

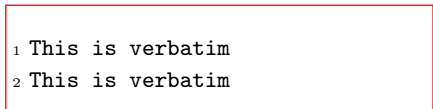
- 带框和标题的摘录

```
\fvset{frame=single,label=摘录,  
      rulecolor=\color{red}}  
\begin{Verbatim}  
This is verbatim  
This is verbatim  
\end{Verbatim}
```



- 带编号的摘录

```
\fvset{numbers=left, numbersep=3pt}  
\begin{Verbatim}  
This is verbatim  
This is verbatim  
\end{Verbatim}
```



# fancyvrb 宏包

## 📖 在摘录前插入命令

```
\fvset{formatcom=\color{blue}\scshape}  
\begin{Verbatim}  
This is verbatim  
This is verbatim  
\end{Verbatim}
```

```
THIS IS VERBATIM  
THIS IS VERBATIM
```

## 📖 更改摘录字体, 大小, 形状, 粗细 (缺省为 `\ttfamily`)

```
\fvset{fontfamily=helvetica,  
      fontsize=\Large,  
      fontshape=it,  
      fontseries=b}  
\begin{Verbatim}  
This is verbatim  
This is verbatim  
\end{Verbatim}
```

```
This is verbatim  
This is verbatim
```

## 📖 更多功能参见该宏包的使用说明

- ① 盒子
- ② 段落盒子与小页环境
- ③ 引文与抄录
- ④ 表格高级技巧**
- ⑤ 罗列
- ⑥ 角注与边注

# 表格高级技巧

## ☞ 表格环境

```
\begin{tabular} [竖向位置] {列格式}
... ..
\end{tabular}
```

```
\begin{tabular*} {宽度} [竖向位置] {列格式}
... ..
\end{tabular*}
```

- 带星号的表格环境可以指定表格的宽度
- 竖向位置: t, b, 缺省垂直居中
- 列格式: l, c, r, —, —, ...

## ☞ 更多表格参数 (列格式)

- p{宽度} → 指定对应列的宽度, 左对齐, 自动换行
- \*{数}{列格式} → 多个相同列格式的缩写

# 表格高级技巧

## ☞ 边界和列间距

- `@{文本}` → `@表达式`, 在对应位置“吃掉”原有的列间隔, 改为插入指定的文本, 若只写 `@{}`, 则表示取消对应位置的列间隔
- 若要改变指定相邻两列的列间距, 可在 `@表达式` 中使用 `\hspace`
- 若在 `@表达式` 中使用 `\extracolsep{宽度}` 命令, 则其后面所有的列间距都增加指定的额外长度

## ☞ 控制表格式样的常用参数

- `\arrayrulewidth` → 各类表格中线条的粗细
- `\doublerulesep` → 双线之间的距离
- `\tabcolsep` → 表格环境中列间距的一半
- `\arraycolsep` → `array` 环境中列间距的一半
  - 以上长度的值可使用 `\setlength` 修改
- `\arraystretch` → 行间距伸缩因子 (可用 `\renewcommand` 修改)
- 例: 62tabular.tex

## 特殊表格

 `longtable` 宏包, 提供 `longtable` 环境, 可实现跨页的长表格


```
\begin{longtable}[横向位置]{列格式}
... ..
\end{longtable}
```

- **横向位置**: l (居左), r (居右), c (居中), 缺省居中
- **列格式**: 与 `tabular` 类似
- 几个有用的命令

```
\endhead → 指定每页都出现在表格最上方的行
\endfirsthead → 指定只出现第一页的表格最上方的行
\endfoot → 指定每页都出现在表格最下方的行
\endlastfoot → 指定只出现最后一页的表格最下方的行
\caption → 表格标题
\pagebreak, \newpage, \noprogrambreak → 分页 / 不准分页
```

- 例: 63longtable.tex

## 特殊表格

-  slashbox 宏包: 提供 `\backslashbox` 命令, 可在表格的单元格中画一条对角线, 线上线上都可以输入文本


```
\backslashbox{对角线下方文本}{对角线上方文本}
```

```
\begin{tabular}{|l|c|c|c|}\hline
\backslashbox{Date}{Room}
& 1 & 2 & 3 \\ \hline
Room 200 & & & \\ \hline
Room 208 & & & \\ \hline
\end{tabular}
```

Room	1	2	3
Date			
Room 200			
Room 208			



## 带颜色的表格

 `colortbl` 宏包: 设置表格文本、行、列、单元格前景背景, 及边框的颜色

```
\columncolor[色系]{颜色名称}[左伸出][右伸出] → 指定某列的颜色
\rowcolor[色系]{颜色名称}[左伸出][右伸出] → 指定某行的颜色
\cellcolor[色系]{颜色名称}[左伸出][右伸出] → 指定单元格的背景颜色
\arrayrulecolor{颜色名称} → 表格框的颜色
\doublerulesepcolor{颜色名称} → 双框线之间空白的颜色
```

- 色系是可选项, 常用有: `rgb`, `gray`, 缺省为 `natural` (例: `63colortbl.tex`)
- `\columncolor` 放在 `tabular` 的列格式中, 同时需要加上 `>{ ... }`, 如:


















```
>{\columncolor{blue}}
```

```
\begin{tabular}%
  {|>{\columncolor[gray]{0.8}}c|c|}
  \hline
  one & two \\ \hline
  three & four \\ \hline
\end{tabular}
```













one	two
three	four

# 颜色名称

## 常用颜色

 <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

 <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>	

- ① 盒子
- ② 段落盒子与小页环境
- ③ 引文与抄录
- ④ 表格高级技巧
- ⑤ 罗列**
- ⑥ 角注与边注

# 罗列

## 常用的三种罗列环境

- `itemize` → 带相同的标签的罗列环境
- `enumerate` → 条目标签为自动编号的符号
- `description` → 人工指定各条目的标签

## 条目输入格式

```
\item[标签] 条目内容
```

- 前两种罗列环境可以省略标签，第三种不能省
- 罗列可以相互嵌套，但每一种罗列的嵌套层数  $\leq 4$

## itemize 罗列环境

### itemize 罗列环境

```
\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
- 各层标签所对应的命令: (可以使用命令 `\renewcommand` 修改)  
`\labelitemi`, `\labelitemii`, `\labelitemiii`, `\labelitemiv`
- 取消某一个条目的标签: `\item[]`
- 例: 64list.tex

## enumerate 罗列环境

### enumerate 罗列环境

```
\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：
  - 阿拉伯数字后跟圆点： 1. 2.
  - 圆括号包围的小写拉丁字母： (a) (b)
  - 小写罗马数字后跟圆点： i. ii.
  - 大写拉丁字母后跟圆点： A. B.
- 各层标签所对应的命令：(可以使用命令 `\renewcommand` 修改)  
`\labelenumi`, `\labelenumii`, `\labelenumiii`, `\labelenumiv`
- 各层缺省标签所对应的计数器为：`enumi`, `enumii`, `enumiii`, `enumiv`
- 指定计数器的值的显示方式：`\arabic`, `\roman`, `\Roman`, `\alph`, `\Alph`
- 选项**标签** → 手工指定标签

## description 罗列环境

---

### description 罗列环境

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

- 该环境没有缺省的标签，只能通过**标签**选项给出，否则就没有标签

# 广义罗列环境

## 广义罗列环境:

```
\begin{list}{标准标签}{罗列声明}
  \item[标签] 条目内容
  ... ..
\end{list}
```

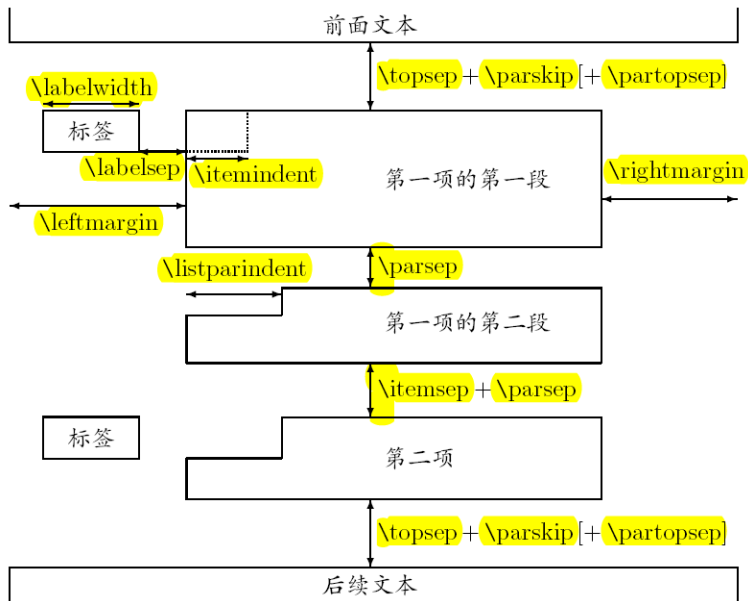
- **标准标签** → 指定缺省标签
- **罗列声明** → 指定罗列的样式, 如: 标签, 条目缩进距离, 段落间距, ...
- 广义罗列环境中的样式参数 (可使用 `\setlength` 等命令修改)

`\topsep`, `\partopsep`, `\itemsep`, `\parsep` `\labelwidth`, `\labelsep`  
`\itemindent`, `\listparindent`, `\leftmargin`, `\rightmargin`

- 标签盒子缩进距离: `\leftmargin`–`\labelwidth`–`\labelsep`
- 若要使用自动编号的标签, 则需要自定义计数器:  
`\newcounter`, `\usecounter` (例: 64list.tex)



# 广义罗列环境样式参数



# 广义罗列环境

## 广义罗列

```
\newcounter{mycounter} % 定义计数器
\begin{list}
  {\hei 问题\, \bfseries\sffamily\arabic{mycounter}: \hfill}
  {\setlength{\parsep}{\parskip}
   \setlength{\itemsep}{0ex plus0.1ex}
   \setlength{\labelwidth}{4em}
   \setlength{\labelsep}{0.2em}
   \setlength{\leftmargin}{6.2em}
   \setlength{\rightmargin}{2em}
   \usecounter{mycounter}
   \setcounter{mycounter}{5}
   \upshape
  }
\item 这是条目内容
\item 这是条目内容
\end{list}
```

# 平凡罗列环境

---

 平凡罗列环境:

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

- 该环境主要用于生成其它环境, 如: `center`, `flushleft`, `flushright`, ...

- ① 盒子
- ② 段落盒子与小页环境
- ③ 引文与抄录
- ④ 表格高级技巧
- ⑤ 罗列
- ⑥ 角注与边注

# 脚注

## 脚注命令

```
\footnote{脚注文本} → 自动编号  
\footnote[标记]{脚注文本} → 手工指定脚注标记
```

- 指定脚注标记的脚注不参加自动编号  
注: 脚注标记只能是整数, 但可以改变其显示样式
- 脚注对应的计数器为 `footnote`
- 修改脚注标记的显示样式, 如: (`\roman`, `\Roman`, `\alph`, `\Alph`, ...)

```
\renewcommand{\thefootnote}{\fnsymbol{footnote}}  
\renewcommand{\thefootnote}{\arabic{footnote}}
```

- 小页环境有独立的脚注计数器 `mpfootnote`  
缺省的显示样式为小写拉丁字母

## 禁止模式中的脚注

- 禁止模式: 不能直接使用脚注命令 `\footnote`  
禁止模式包括: 数学环境, 表格, LR 盒子, 段落盒子

- 在禁止模式中使用脚注的方法


- 分两步实现: 做脚注标记 + 写脚注文本

```
\footnotemark → 脚注标记命令  
\footnotetext{脚注文本} → 脚注文本命令
```

```
\footnotemark[标记] → 手工指定脚注标记  
\footnotetext[标记]{脚注文本}
```

- `\footnotemark` 位于禁止模式中需要注释的地方  
`\footnotetext` 位于禁止模式之外, 最好紧跟在禁止模式后面
- 如果手工指定脚注标记, 则同一脚注必须使用统一的标记
- 例: 65footnote.tex

## 特殊角注


 `\footnotemark` 与 `\footnotetext` 的匹配 (自动编号时)

- 每调用一次 `\footnotemark`, 计数器 `footnote` 的值就会自动加 1
- `\footnotetext` 只使用 `footnote` 值, 不改变其值
- 当禁止模式中含多个脚注时, 在禁止模式外使用 `\footnotetext` 命令时需手工调整 `footnote` 值, 同时每用一次脚注文本命令后, 需增加脚注计数器的值 (最后一次调用不需要)

```
\addtocounter{footnote}{负数}  
\stepcounter{footnote}
```

- 特殊效果: 多个连续脚注使用同一个脚注标记

# 边注


 边注: 在页面边上添加注释

```
\marginpar{边注文本}
```

- 边注为一个较窄的段落盒子 → [边注盒子](#)
- 边注缺省出现在页面右侧; 若使用双面选项(`twoside`), 则奇右偶左; 若使用双列选项(`twocolumn`), 则左左右右  
`\reversemarginpar` → 改变边注的缺省位置
- 控制边注样式的常用参数 (用 `\setlength` 修改)

```
\marginparwidth → 边注盒子宽度
```

```
\marginparsep → 边注盒子与正文之间的距离
```

 book 类文档的边注的特殊处理方法

```
\marginpar[\flushright 左边边注]{右边边注}
```

- 若边注出现在奇数页, 则输出右边注, 并位于页面右边;  
若边注出现在偶数页, 则输出左边注, 并位于页面左边