



我的 LaTeX 课程

入门学习笔记

作者：雨霓同学

邮箱：910014191@qq.com

时间：April 24, 2020

组织：苓

QQ：910014191



别人都关心你飞的有多高，只有我关心你的翅膀好不好吃！

目录

1	LaTeX 入门之公式篇	2
1.1	数学模式	2
1.1.1	行内公式	2
1.1.2	行间公式	2
1.2	常用数学宏包	3
1.2.1	花式数学字体	3
1.2.2	公式字体加粗	3
1.3	公式的编号	4
1.3.1	公式的自动编号	4
1.3.2	使用标签自定义编号	4
1.3.3	公式的重新编号	4
1.3.4	公式的子编号	5
1.3.5	其他补充编号	5
1.4	amsmath 的公式环境	8
1.4.1	equation 环境	8
1.4.2	align, align*, aligned 环境	8
1.4.3	alignat, alignat*, alignedat 环境	9
1.4.4	gather, gather*, gathered 环境	10
1.4.5	flalign, multiline	10
1.4.6	split 环境	11
1.4.7	array, eqnarray 环境	11
1.4.8	单花括号环境	12
1.5	定界符宏包	13
1.6	矩阵环境	13
1.6.1	array 环境	13
1.6.2	matrix 环境	14
1.6.3	bmatrix 环境	15
1.6.4	Bmatrix 环境	15
1.6.5	pmatrix 环境	15
1.6.6	vmatrix 环境	16
1.6.7	Vmatrix 环境	16
2	LaTeX 入门之表格篇	17
2.1	常用表格环境 tabular	17
2.1.1	垂直对齐方式设置	17
2.1.2	列格式说明	18
2.1.3	其他参数	19
2.2	跨行表格宏包 multirow	20
2.3	表格的浮动环境	21

2.3.1	位置参数	21
2.3.2	示例	21
2.4	三线表	22
2.5	跨页长表格	22
2.6	其他宏包与表格缩放	24
2.7	列表的使用	25
2.7.1	无序列表	25
2.7.2	有序列表	26
2.7.3	解说列表	28
2.7.4	通用列表环境 list	29
2.7.5	列表宏包 enumitem	29
3	LaTeX 入门之插图篇	30
3.1	图形的分类	30
3.1.1	位图图像	30
3.1.2	向量图形	30
3.1.3	插图的基本命令	30
3.2	插图命令的参数	31
3.2.1	外形参数说明	31
3.2.2	关于图片的高度与深度	31
3.2.3	设置图片的缩放	32
3.2.4	关于图片的旋转	32
3.3	图片的标题和浮动	33
3.4	多个插图放置	34
3.4.1	两图并列, 共享标题	34
3.4.2	两图并列, 各有标题, 共享标题	35
3.4.3	其他插图	37
3.5	eps 图像获取	40
4	LaTeX 入门之参考文献篇	41
4.1	参考文献	41
4.1.1	什么是参考文献	41
4.1.2	基本的参考文献	41
4.1.3	参考文献的引用	42
4.1.4	引文格式修改	42
4.1.5	bibtex 文献管理使用	43
4.2	交叉引用	43
4.3	网址链接	43
4.4	代码框设计	44

特别声明

在此特别感谢, [Elegant \$\text{\LaTeX}\$](#) 系列模板的各位作者大大, 没有他们我去哪里找这么好看的模板。这是模板的一些地址: [GitHub](#)、[CTAN](#)、[Overleaf](#) 以及 [Gitee](#) 感兴趣的同学的可以自己下载!

关于为什么有这个内容的存在? emmmm 卤煮大概是大三的时候学习的 \LaTeX 。平时主要用这个排版一些学习笔记, 后来大大小小的学术期刊、论文都排过一些, 所以慢慢就想着把自己学习 \LaTeX 的内容记录一下, 于是就有了这个玩意! 希望能够帮助到想要学习或者入门 \LaTeX 的同学!

如果你发现我写的内容, 那些地方有问题, 希望可以留言给我, 联系方式在封面页! 这里会尽可能的改正! 目前还有一些 Bug 没能解决, 希望有能力的大佬帮助解决下! (第一处笔记处有记录)

梁霄

April 24, 2020

第一章 LaTeX 入门之公式篇

内容提要

- ❑ 数学模式
- ❑ 常用数学宏包
- ❑ 公式的编号
- ❑ 公式环境
- ❑ 定界符宏包
- ❑ 矩阵环境

1.1 数学模式

1.1.1 行内公式

LaTeX 提供三种方法来编写行内公式

1. `$...这是公式内容...$`
2. `\(...这是公式内容...\)`
3. `\begin{math}...这是公式内容...\end{math}`

如有如下示例：

四叶线玫瑰线 $p = a \sin 2\beta$ 方程

四叶线玫瑰线 `\(p=a\sin2\beta\)` 方程

四叶线玫瑰线

`\begin{math}p=a\sin2\beta\end{math}` 方程

四叶线玫瑰线 $p = a \sin 2\beta$ 方程

四叶线玫瑰线 $p = a \sin 2\beta$ 方程

四叶线玫瑰线 $p = a \sin 2\beta$ 方程



笔记 常用的行内数学模式效果是一样的，不过第一种方法是比较常用的，原因嘛，使用方便，但是缺点是不能区分起止，还有一种原因是因为它性格坚强（是因为它能出现在一些其他命令中，像上面的 `\verb|text|` 这就是一个脆弱命令，它不能直接出现在任何命令中。（\$ 这个符号的公式显示貌似有点 bug，代码区域不显示 \$ 符号）

1.1.2 行间公式

LaTeX 提供三种方法来编写行内公式

1. `$$...这是公式内容...$$`
2. `\[...这是公式内容...\]`
3. `\begin{displaymath}...这是公式内容...\end{displaymath}`

如有如下示例：

四叶线玫瑰线: $p=a\sin 2\beta$ 方程.	四叶线玫瑰线:
四叶线玫瑰线: $\backslash[p=a\sin 2\beta]$ 方程.	$p = a \sin 2\beta$
四叶线玫瑰线:	方程.
$\backslash\begin{displaymath}$	四叶线玫瑰线:
$p=a\sin 2\beta$	$p = a \sin 2\beta$
$\backslash\end{displaymath}$	方程.
方程.	四叶线玫瑰线:
	$p = a \sin 2\beta$
	方程.



笔记 常用的行内数学模式效果是一样的,也是第一种方法是比较常用,这是 $\text{T}_{\text{E}}\text{X}$ 的原始方法,只有在极个别的情况下会出现问题 (`fleqn`) 环境中,在数学模式中一般不建议出现空行或者 `\par` 等换行命令,因为系统会提示出错。

1.2 常用数学宏包

1.2.1 花式数学字体

常见公式宏包 `amsmath`、`mathrsfs`、`amsfonts`、`bm` 宏包导言区使用 `\usepackage{amsmath}`

$\backslash\mathrm{ABCDEF GHI}$	<i>A B C D E F G H I</i>
$\backslash\mathcal{ABCDEF GHI}$	<i>A B C D E F G H I</i>
$\backslash\mathbb{ABCDEF GHI}$	A B C D E F G H I
$\backslash\mathfrak{ABXDEF GHI}$	A B X D E F G H I

1.2.2 公式字体加粗

$\backslash\mathrm{X^2+Y^2=Z^2}$	$\mathbf{X^2 + Y^2 = Z^2}$
$\backslash\mathbf{\mathrm{X^2+Y^2=Z^2}}$	$\mathbf{X^2 + Y^2 = Z^2}$
$\mathrm{X^2+Y^2=Z^2}$	$X^2 + Y^2 = Z^2$

这一部分通常不太用,看需求吧,如果你有花里胡哨的需求,可以参考 `symbols-a4.pdf` 这本书,里面各种花里胡哨的符号都有,很强大,书籍地址:[提取码:cycb](#) [点击跳转](#)

1.3 公式的编号

1.3.1 公式的自动编号

一般情况都是行间公式才会进行公式编号，那么如何进行编号呢？

这是一个行间公式，我们可以使用

```
\begin{equation}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
```

进行行间公式编号

这是一个行间公式，我们可以使用

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (1.3.1)$$

进行行间公式编号

1.3.2 使用标签自定义编号

当然也可以使用一些标签进行自定义编号，一般很少这样子用，示例如下：

这是一个行间公式，我们可以使用 `\verb|\tag{*}|`

```
\begin{equation}\tag{*}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
```

进行行间公式标签自定义编号

这是一个行间公式，我们可以使用 `\tag{*}`

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (*)$$

进行行间公式标签自定义编号

1.3.3 公式的重新编号

我们可以使用自定义命令，让原本按章编号的公式现在按节进行编号，这个准确的名称叫做公式的排序单位

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

这是按节编号

```
\begin{equation}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
```

进行行间公式编号

这是按节编号

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (1.3.2)$$

进行行间公式编号

```
\numberwithin{equation}{chapter}
```

这里是按章进行编号，注意看

```
\begin{equation}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
```

这里是按章进行编号，注意看

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (1.3)$$



笔记 我们可以通过更改 `\numberwithin{equation}{section}` 中的 `section` 选项来改变编号形式，这里默认是 `section`。

1.3.4 公式的子编号

有时候一个行间公式有多个子公式，这时候需要对每个子公式进行编号，可以这样子去做。

这是多公式进行子编号

```
\begin{subequations}
\begin{equation}\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx
\end{equation}
\begin{equation}\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx
\end{equation}
\end{subequations}
```

这是多公式进行子编号

$$\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx \quad (1.3.4a)$$

$$\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx \quad (1.3.4b)$$

同样的我们也可以使用标签进行自定义命令，

```
\begin{subequations}
\begin{equation}\tag{\alpha}\underset{x \rightarrow 0}{\lim} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
\begin{equation}\tag{\beta}\underset{x \rightarrow 0}{\lim} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1
\end{equation}
\end{subequations}
```

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (\alpha)$$

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (\beta)$$

同样的也可以使用汉字进行编号

```
\begin{subequations}
\begin{equation}\tag{公式 1}\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx
\end{equation}
\begin{equation}\tag{公式 2}\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx
\end{equation}
\end{subequations}
```

$$\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx \quad (\text{公式 1})$$

$$\lim_{x \rightarrow 0} \frac{1}{bx - \sin x} dx \quad (\text{公式 2})$$

1.3.5 其他补充编号

还有一些其他的编号方式，这里做下整理，也可以参照示例：



<code>\eqno{标号}</code>	系统提供的一个序号设置命令，可以放在 <code>equation*</code> 或 <code>\[...]</code> 形式的公式行后，可以在公式右侧人工设置编号，标号是任意文本
<code>\leqno{标号}</code>	作用与 <code>\eqno{标号}</code> 相似，只是将标号置于公式的左侧。对于两个不能同时使用，也不会出现这样奇葩的需求吧！
<code>\nonumber</code>	系统提供的取消序号命令，可以把它插在换行命令之前，就可以为取消改行公式的序号
<code>\notag</code>	和 <code>\nonumber</code> 是一样的
<code>\tag*{标号}</code>	和 <code>\tag{label}</code> 相同，只是标号两侧没有圆括号

使用 `\verb|\eqno{标号}|`

```
\begin{equation*}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt
\tag*{\text{公式1}}
\end{equation*}
```

使用 `\eqno{标号}`

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad \text{公式 1}$$

使用 `\verb|\leqno{标号}|`

```
\begin{equation*}
\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt
\leqno{\text{公式1}}
\end{equation*}
```

使用 `\leqno{标号}`

$$\text{公式 1} \quad \lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1$$

使用 `\verb|\nonumber|`

```
\begin{align}
(a+b)^4 &= (a+b)^2(a+b)^2 \\
&= \left(a^2+2a\right) \left(a^2+2a\right) \\
&= a^4+4a^3b+6a^2b^2+4ab^3+b^4
\end{align}
```

使用 `\nonumber`

$$\begin{aligned} (a+b)^4 &= (a+b)^2(a+b)^2 \\ &= (a^2+2ab+b^2)(a^2+2ab+b^2) \\ &= a^4+4a^3b+6a^2b^2+4ab^3+b^4 \end{aligned} \quad (1.3.7)$$

使用 `\verb|\notag|`

```
\begin{align}
(a+b)^4 &= (a+b)^2(a+b)^2 \\
&= \left(a^2+2a\right) \left(a^2+2a\right) \\
&= a^4+4a^3b+6a^2b^2+4ab^3+b^4
\end{align}
```

使用 `\notag`

$$\begin{aligned} (a+b)^4 &= (a+b)^2(a+b)^2 \\ &= (a^2+2ab+b^2)(a^2+2ab+b^2) \\ &= a^4+4a^3b+6a^2b^2+4ab^3+b^4 \end{aligned} \quad (1.3.8)$$

使用 `\verb|\tag*{标号}|`
`\begin{equation}`
`\lim_{x \rightarrow 0} \int_0^x`
`\frac{t^2}{\sqrt{a+t^2}} dt`
`t=1\tag*{\text{公式1}}`
`\end{equation}`

使用 `\tag*{标号}`

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad \text{公式 1}$$

使用 `\verb|\tag{标号}|`
`\begin{equation}`
`\lim_{x \rightarrow 0} \int_0^x`
`\frac{t^2}{\sqrt{a+t^2}} dt`
`t=1\tag{\text{公式1}}`
`\end{equation}`

使用 `\tag{标号}`

$$\lim_{x \rightarrow 0} \int_0^x \frac{t^2}{\sqrt{a+t^2}} dt = 1 \quad (\text{公式 1})$$



1.4 amsmath 的公式环境

公式宏包 amsmath 是由美国数学学会组织编写的一个公式宏包，它是 ams 宏包套件中最主要的宏包，现在编写行间公式通常采用 amsmath 提供的各种公式环境。具体如下表所示：

表 1.1: 公式宏包 amsmath 提供的各种公式环境

环境名	用途	环境名	用途
equation	单行公式环境	equation*	单行公式环境，无序号
align	公式组环境	align*	公式组环境，无序号
alignat	公式组环境	alignat*	公式组环境，无序号
flalign	公式组环境	flalign*	公式组环境，无序号
gather	公式组环境	gather*	公式组环境，无序号
aligned	块环境，无序号	alignedat	块环境，无序号
gathered	块环境，无序号，		
multline	多行公式环境	multline*	多行公式环境，无序号
split	多行公式环境，无序号		
cases	左花括号环境，无序号	subequations	子公式环境

具体示例如下：

1.4.1 equation 环境

单行公式环境 equation 无论公式多长，都可以排为一行，并给出一个序号，使用 equation* 则可以去掉序号。

单行公式环境 equation $\begin{aligned} &\text{\textcolor{red}{\backslash begin}\{equation\}} \\ &f(x)=3x^2+6(x-2)-1 \\ &\text{\textcolor{red}{\backslash end}\{equation\}} \\ &\text{\textcolor{red}{\backslash vskip 2mm \hrule \vskip 2mm}} \\ &\text{单行公式环境 equation*} \\ &\text{\textcolor{red}{\backslash begin}\{equation*\}} \\ &f(x)=3x^2+6(x-2)-1 \\ &\text{\textcolor{red}{\backslash end}\{equation*\}} \end{aligned}$	单行公式环境 equation $f(x) = 3x^2 + 6(x - 2) - 1 \quad (1.4.1)$ <hr/> 单行公式环境 equation* $f(x) = 3x^2 + 6(x - 2) - 1$
--	--

1.4.2 align, align*, aligned 环境

如果要求共十组或多行公式以其中某个符号对齐，可以 align 环境环境，它以\\为分行符，每行都输出序号，它以&为分列标志，奇数列右对齐，偶数列左对齐，奇偶列并肩对齐

这是align环境 $\begin{aligned} &\text{\textcolor{red}{\backslash begin}\{align\}} \\ &f(x) \ \&= 2(x+1)^2-1 \ \backslash \\ &\&= 2 \ x^2+4 \ x+1 \\ &\text{\textcolor{red}{\backslash end}\{align\}} \end{aligned}$	这是 align 环境 $f(x) = 2(x + 1)^2 - 1 \quad (1.4.2)$ $= 2x^2 + 4x + 1 \quad (1.4.3)$
--	--

这是align*环境，不需要数学模式

```
\begin{align*}
f(x) &= 2(x+1)^2 - 1 \\
&= 2\left(x^2 + 2x + 1\right) - 1 \\
&= 2x^2 + 4x + 1
\end{align*}
```

这是 align* 环境，不需要数学模式

$$\begin{aligned} f(x) &= 2(x+1)^2 - 1 \\ &= 2(x^2 + 2x + 1) - 1 \\ &= 2x^2 + 4x + 1 \end{aligned}$$

aligned环境需要数学模式

```
\begin{aligned}
f(x) &= 2(x+1)^2 - 1 \\
&= 2\left(x^2 + 2x + 1\right) - 1 \\
&= 2x^2 + 4x + 1
\end{aligned}
```

aligned 环境需要数学模式

$$\begin{aligned} f(x) &= 2(x+1)^2 - 1 \\ &= 2(x^2 + 2x + 1) - 1 \\ &= 2x^2 + 4x + 1 \end{aligned}$$

1.4.3 alignat, alignat*, alignedat 环境

公式组 alignat 环境与 align 环境相似，不同之处在于列对之间的默认距离为 0pt，通常要插入水平空白命令，用于控制列对之间的距离，此外该环境还有一个参数项，用于设置列的个数，其数目需大于实际值，但不能小于。

alignat环境不需要数学模式

```
\begin{alignat}{3}
A_{1}&=B_{1} \quad B_{2} \quad \backslash quad \\
&\quad \&A_{3}&=B_{1} \\
A_{2}&=B_{3} \quad \&A_{3} \\
&\quad A_{4}&=B_{4}
\end{alignat}
```

alignat 环境不需要数学模式

$$A_1 = B_1 B_2 \quad A_3 = B_1 \quad (1.4.4)$$

$$A_2 = B_3 \quad A_3 A_4 = B_4 \quad (1.4.5)$$

alignat环境不需要数学模式

```
\begin{alignat*}{3}
A_{1}&=B_{1} \quad B_{2} \quad \backslash quad \\
&\quad \&A_{3}&=B_{1} \\
A_{2}&=B_{3} \quad \&A_{3} \\
&\quad A_{4}&=B_{4}
\end{alignat*}
```

alignat 环境不需要数学模式

$$A_1 = B_1 B_2 \quad A_3 = B_1$$

$$A_2 = B_3 \quad A_3 A_4 = B_4$$

alignedat环境需要数学模式

```
\begin{alignedat}{3}
A_{1}&=B_{1} \quad B_{2} \quad \backslash quad \\
&\quad \&A_{3}&=B_{1} \\
A_{2}&=B_{3} \quad \&A_{3} \\
&\quad A_{4}&=B_{4}
\end{alignedat}
```

alignedat 环境需要数学模式

$$A_1 = B_1 B_2 \quad A_3 = B_1$$

$$A_2 = B_3 \quad A_3 A_4 = B_4$$

1.4.4 gather, gather*, gathered 环境

gather 用于编写中心对称的公式组，使用换行命令来区分各个命令，每个公式都与公式行居中对齐。

gather环境不需要数学模式 $\begin{gathered} \ln y = x \ln x \\ y' = x^x(1 + \ln x) \end{gathered}$	gather 环境不需要数学模式 $\ln y = x \ln x \quad (1.4.6)$ $y' = x^x(1 + \ln x) \quad (1.4.7)$
gather*环境不需要数学模式 $\begin{gathered} \ln y = x \ln x \\ y' = x^x(1 + \ln x) \end{gathered}$	gather* 环境不需要数学模式 $\ln y = x \ln x$ $y' = x^x(1 + \ln x)$
gathered环境需要数学模式 $\begin{gathered} \ln y = x \ln x \\ y' = x^x(1 + \ln x) \end{gathered}$	gathered 环境需要数学模式 $\ln y = x \ln x$ $y' = x^x(1 + \ln x)$

1.4.5 flalign, multiline

flalign 与 align 的功能基本相同，唯一区别是列对之间的距离为弹性宽度，以使公式组两端对齐


flalign环境不需要数学模式 $\begin{aligned} A_1 &= B_1 B_2 \\ A_3 &= B_1 \\ A_2 &= B_3 \\ A_3 A_4 &= B_4 \end{aligned}$	flalign 环境不需要数学模式 $3A_1 = B_1 B_2 \quad A_3 = B_1 \quad (1.4.8)$ $A_2 = B_3 \quad A_3 A_4 = B_4 \quad (1.4.9)$
flalign环境不需要数学模式 $\begin{aligned} A_1 &= B_1 B_2 \\ A_3 &= B_1 \\ A_2 &= B_3 \\ A_3 A_4 &= B_4 \end{aligned}$	flalign 环境不需要数学模式 $3A_1 = B_1 B_2 \quad A_3 = B_1$ $A_2 = B_3 \quad A_3 A_4 = B_4$

多行公式环境 multlins 可用于编写多行公式，首航左对齐，尾行右对齐，中间居中对齐，公式编号在行尾给出。

<p>multline环境不需要数学模式</p> <pre>\begin{multline} A_{\{m\}}^{\{n\}}=m(m-1) \\ +(m-2) \\ \\+[m-(n-1)] \\ \end{multline}</pre>	<p>multline 环境不需要数学模式</p> $A_m^n = m(m-1) + (m-2) + [m-(n-1)] \quad (1.4.10)$
<pre>\begin{multline*} A_{\{m\}}^{\{n\}}=m(m-1) \\ +(m-2) \\ \\+[m-(n-1)] \\ \end{multline*}</pre>	$A_m^n = m(m-1) + (m-2) + [m-(n-1)]$

使用\shoveleft,\shoveright可以使中间公式左对齐或者右对齐

<pre>\begin{multline} A_{\{m\}}^{\{n\}}=m(m-1) \\ \shoveleft{+(m-2)} \\ +[m-(n-1)] \\ \end{multline}</pre>	$A_m^n = m(m-1) + (m-2) + [m-(n-1)] \quad (1.4.11)$
<pre>\begin{multline*} A_{\{m\}}^{\{n\}}=m(m-1) \\ \shoveright{+(m-2)} \\ +[m-(n-1)] \\ \end{multline*}</pre>	$A_m^n = m(m-1) + (m-2) + [m-(n-1)]$

 **笔记** 这个总感觉不太好用

1.4.6 split 环境

split 也用于排版多行公式，它与 multline 的区别主要是以下三点。

1. 使用&作为分隔符，但至多两列，左列右对齐，右列左对齐，如果不使用分隔符，则所有公式行为一列，且全部与首航公式右端对齐
2. 必须置身于除 multline 环境之外的其他公式环境中
3. 自身并不生成公式序号，而是由外在公式环境提供，序号垂直居中

<pre>\begin{equation} \\ \begin{split} f(x) &= 2(x+1)^2 - 1 \\ &= 2\left(x^2 + 2x + 1\right) - 1 \\ &= 2x^2 + 4x + 1 \end{split} \\ \end{equation}</pre>	$f(x) = 2(x+1)^2 - 1 = 2(x^2 + 2x + 1) - 1 = 2x^2 + 4x + 1 \quad (1.4.12)$
--	--

1.4.7 array,eqnarray 环境

数组宏包 array 对系统提供的数组环境 array 做了功能扩展，其参数位置和列格式的各种选项及功能与表格环境完全相同

<pre> \begin{equation*} \left. \begin{array}{l} \text{常数)} \ y = c \\ \text{直线)} \ y = cx + d \\ \text{抛物线)} \ y = bx^2 + cx + d \end{array} \right\} \text{多项式} \end{equation*} </pre>	$\left. \begin{array}{l} \text{常数)} \ y = c \\ \text{直线)} \ y = cx + d \\ \text{抛物线)} \ y = bx^2 + cx + d \end{array} \right\} \text{多项式}$
---	--

eqnarray 公式组环境是由系统提供的，其默认公式对齐方式为右对齐。

<pre> \begin{eqnarray} E = \hbar \cdot \nu \\ E = m \cdot c^2 \end{eqnarray} </pre>	$E = \hbar \cdot \nu \quad (1.4.13)$ $E = m \cdot c^2 \quad (1.4.14)$
---	---

<pre> \begin{eqnarray} E = \hbar \cdot \nu \\ E = m \cdot c^2 \end{eqnarray} </pre>	$E = \hbar \cdot \nu \quad (1.4.15)$ $E = m \cdot c^2 \quad (1.4.16)$
---	---

1.4.8 单花括号环境

1.4.8.1 左花括号环境

左花括号环境 cases 用于在其他公式环境中排版带有左花括号的公式

<pre> \begin{equation} x = \begin{cases} x & \text{如果 } x \geq 0 \\ -x & \text{如果 } x \leq 0 \end{cases} \end{equation} </pre>	$ x = \begin{cases} x & \text{如果 } x \geq 0 \\ -x & \text{如果 } x \leq 0 \end{cases} \quad (1.4.17)$
--	--

<pre> \begin{subnumcases}{ x = } x & x \geq 0 \\ -x & \leq 0 \end{subnumcases} </pre>	$ x = \begin{cases} x & x \geq 0 \quad (1.4.18a) \\ -x & \leq 0 \quad (1.4.18b) \end{cases}$
--	---

<pre> \begin{numcases}{ x = } x & x \geq 0 \\ -x & \leq 0 \end{numcases} </pre>	$ x = \begin{cases} x & x \geq 0 \quad (1.4.19) \\ -x & \leq 0 \quad (1.4.20) \end{cases}$
--	---

1.4.8.2 右花括号环境

右花括号环境 `rcases` 用于在其他公式环境中排版带有左花括号的公式

<pre>\begin{equation*} \begin{rcases} \text{正无理数} \\ \text{负无理数} \end{rcases} \text{无限不循环小数} \end{equation*}</pre>	$\left. \begin{array}{l} \text{正无理数} \\ \text{负无理数} \end{array} \right\} \text{无限不循环小数}$
--	--

1.5 定界符宏包

给带定界符的方程组的每一行都编号, 这种情况 `amsmath` 包无法实现, 我们可以用 `cases` 包的 `numcases` 环境

<pre>%\usepackage{cases} \begin{numcases}{f(x)=}%f(x)=可以置空 1,&x \in \mathbb{Q} \\ 0,&x \notin \mathbb{Q} \end{numcases}.</pre>	$f(x) = \begin{cases} 1, & x \in \mathbb{Q} & (1.5.1) \\ 0, & x \notin \mathbb{Q} & (1.5.2) \end{cases}$
--	--

不过上述 `numcases` 环境的效果是不尽如人意的, 更好的效果是用 `empheq` 包, 它可以给 `amsmath` 包提供的数学环境添加各种定界符.

<pre>\begin{empheq}[left=\empheqlbrace, right=\empheqrbrack]{align} &a=b+c&&a=b \\ &a^2=b^2+c^2&&a=b \end{empheq}</pre>	$\left[\begin{array}{ll} a = b + c & a = b \end{array} \right] & (1.5.3)$ $\left[\begin{array}{ll} a^2 = b^2 + c^2 & a = b \end{array} \right] & (1.5.4)$
---	---

1.6 矩阵环境

常用的矩阵输入, 可以使用 `array`、`matrix`、`Bmatrix`、`bmatrix`、`Pmatrix`、`vmatrix`、`Vmatrix` 等等

1.6.1 `array` 环境

需要使用数学模式

<pre>\left[\begin{array}{ccc} 1&2&3 \\ 4&5&6 \\ 7&8&9 \end{array} \right]</pre>	$\left[\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right]$
--	---

```

\left\{
\begin{array}{ccc}
1&2&3 \\
4&5&6 \\
7&8&9
\end{array}
\right\}

```

$$\left\{ \begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right\}$$

```

\left|
\begin{array}{ccc}
1&2&3 \\
4&5&6 \\
7&8&9
\end{array}
\right|

```

$$\left| \begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right|$$

```

\left(
\begin{array}{ccc}
1&2&3 \\
4&5&6 \\
7&8&9
\end{array}
\right)

```

$$\left(\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right)$$

1.6.2 matrix 环境

需要使用数学模式

```

\begin{matrix}
1&2&3 \\
4&5&6 \\
7&8&9
\end{matrix}

```

$$\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix}$$

<pre> \left[\begin{matrix} 1&2&3 \\ 4&5&6 \\ 7&8&9 \end{matrix} \right] </pre>	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
---	---

1.6.3 bmatrix 环境

<pre> \begin{bmatrix} 1&2&3 \\ 4&5&6 \\ 7&8&9 \end{bmatrix} </pre>	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
--	---

1.6.4 Bmatrix 环境

<pre> \begin{Bmatrix} 1&2&3 \\ 4&5&6 \\ 7&8&9 \end{Bmatrix} </pre>	$\begin{Bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{Bmatrix}$
--	---

1.6.5 pmatrix 环境

<pre> \begin{equation}\tag{***} \begin{pmatrix} 1&2&3 \\ 4&5&6 \\ 7&8&9 \end{pmatrix} \end{equation} </pre>	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \tag{***}$
---	---

1.6.6 vmatrix 环境

```

\begin{equation}\tag{***}
\begin{vmatrix}
1&2&3 \\
4&5&6 \\
7&8&9
\end{vmatrix}
\end{equation}

```

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \quad (***)$$

1.6.7 Vmatrix 环境

```

\begin{Vmatrix}
1&2&3 \\
4&5&6 \\
7&8&9
\end{Vmatrix}
\leqno{(\text{公式1})}
这是公式\ref{ces}

```

(公式 1)

$$\left\| \begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix} \right\|$$

这是公式***

```

\[
\left[\begin{array}{cccc}
w_{11}&w_{12}&\cdots&w_{1n} \\
w_{21}&w_{22}&\ldots&w_{2n} \\
\cdots&\sim&\sim&\sim \\
\cdots&\sim&\sim&\sim \\
w_{n1}&w_{n2}&\ldots&w_{nn}
\end{array}\right]
\]

```

$$\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \cdot & & & \\ \cdot & & & \\ w_{n1} & w_{n2} & \cdots & w_{nn} \end{bmatrix}$$

第二章 LaTeX 入门之表格篇

内容提要

- 常用表格环境
- 跨行表格
- 表格浮动
- 三线表跨页表
- 其他表格宏包
- 列表的使用

2.1 常用表格环境 tabular

LaTeX 里排版表格不如 Word 等所见即所得的工具简便和自由，不过对于不太复杂的表格来讲，完全能够胜任。

数组宏包 array 改进了扩展了 LaTeX 的 tabular, tabular* 和 arraya 的功能环境，主要是增强了列格式功能，还添加了许多表格参数的功能调整，array 和 tabular 环境相似。

排版表格最基本的 tabular 环境用法为：

```
1 \begin{tabular}[<垂直对齐方式>]{<列格式说明>}
2 <单元格内容> & ... & <单元格内容> \\
3 .....
4 <单元格内容> & ... & <单元格内容> \\
5 \end{tabular}
```

2.1.1 垂直对齐方式设置

- t：表格顶线与当前文本行的基线对齐
- c：默认参数，表格中线与当前文本行的基线对齐
- b：表格底线与当前文本行的基线对齐
- \fisthline：更改 t 参数的对齐方式
- \lasthline：更改 b 参数的对齐方式

这是文本行：

```
\begin{tabular}[c]{|l|l|} \hline
<1> & <2> \\ \hline
<3> & <4> \\ \hline
\end{tabular}
```

\vskip 3mm

这是文本行：

```
\begin{tabular}[t]{|l|l|} \hline
<1> & <2> \\ \hline
<3> & <4> \\ \hline
\end{tabular}
```

这是文本行：

<1>	<2>
<3>	<4>

这是文本行：

<1>	<2>
<3>	<4>

这是文本行:

```
\begin{tabular}[b]{|l|l|} \hline
<1> & <2> \\ \hline
<3> & <4> \\ \hline
\end{tabular}
```

\vskip 3mm

这是文本行:

```
\begin{tabular}[b]{|l|l|} \hline
<1> & <2> \\ \hline
<3> & <4> \\ \hline
\end{tabular}
```

\vskip 3mm

这是文本行:

```
\begin{tabular}[t]{|l|l|} \firsthline
<1> & <2> \\ \hline
<3> & <4> \\ \hline
\end{tabular}
```

这是文本行:

<1>	<2>
<3>	<4>

这是文本行:

<1>	<2>
<3>	<4>

这是文本行:

<1>	<2>
<3>	<4>

2.1.2 列格式说明

- l: 左对齐
- c: 居中对齐
- r: 右对齐
- |: 画纵向的线

1. 左对齐 \par

```
\begin{tabular}{|l|l|l|} \hline
\bfseries left & \bfseries center & \bfseries
right \\ \hline
```

左对齐&居中对齐&右对齐 \\ \hline

\end{tabular} \par

2. 居中对齐 \par

```
\begin{tabular}{|c|l|l|} \hline
\bfseries left & \bfseries center & \bfseries
right \\ \hline
```

左对齐&居中对齐&右对齐 \\ \hline

\end{tabular} \par

3.1. 居中对齐 \par

```
\begin{tabular}{|r|r|r|} \hline %*{3}{|r|}
\bfseries left & \bfseries center & \bfseries
right \\ \hline
```

左对齐&居中对齐&右对齐 \\ \hline

\end{tabular}

1. 左对齐

left	center	right
左对齐	居中对齐	右对齐

2. 居中对齐

left	center	right
左对齐	居中对齐	右对齐

3.1. 居中对齐

left	center	right
左对齐	居中对齐	右对齐

2.1.3 其他参数

1. $*\{n\}\{\text{列格式}\}$ 表示 n 个列格式选项相同的相邻列，例如 $\{3\}\{c\}$ ，它表示表格中相邻三列的列格式是相同的；每列数据居中排列并且两侧都有一条垂直线，它等效于列格式： $|c|c|c|$ 。
 2. $p\{\text{宽度}\}$ 设定该列所占的宽度，并且该列数据左对齐排列，垂直方向顶端对齐。
 3. $m\{\text{宽度}\}$ 设定该列所占的宽度，并且列数据左对齐排列，垂直方向中心对齐
 4. $b\{\text{宽度}\}$ 设定该列所占的宽度，并且该列数据左对齐排列，垂直方向底端对齐
- 示例如下：

```

1 p\{宽度\}
2
3 \begin{tabular}{|p{3cm}|p{3cm}|p{5cm}|} \hline
4 <单元格> & <单元格内容单元格内>& <单元格> \\ \hline
5 <单元格> & <单元格内容><单元格内容> & <单元格> \\ \hline
6 \end{tabular}
7
8 m\{宽度\}
9
10 \begin{tabular}{|m{3cm}|m{3cm}|m{5cm}|} \hline
11 <单元格> & <单元格内容单元格内>& <单元格> \\ \hline
12 <单元格> & <单元格内容><单元格内容> & <单元格> \\ \hline
13 \end{tabular}
14
15 b\{宽度\}
16
17 \begin{tabular}{|b{3cm}|b{3cm}|b{5cm}|} \hline
18 <单元格> & <单元格内容单元格内>& <单元格> \\ \hline
19 <单元格> & <单元格内容><单元格内容> & <单元格> \\ \hline
20 \end{tabular}

```

1. $p\{\text{宽度}\}$

< 单元格 >	< 单元格内容单元 格内 >	< 单元格 >
< 单元格 >	< 单元格内容 >< 单元格内容 >	< 单元格 >

2. $m\{\text{宽度}\}$

< 单元格 >	< 单元格内容单元 格内 >	< 单元格 >
< 单元格 >	< 单元格内容 >< 单元格内容 >	< 单元格 >

3. $b\{\text{宽度}\}$

< 单元格 >	< 单元格内容单元 格内 >	< 单元格 >
< 单元格 >	< 单元格内容 >< 单元格内容 >	< 单元格 >



笔记 `\hline`: 该命令必须用于首行之前或者紧跟在换行命令`\\`之后, 它表示画一条长度与表格宽度相同的水平线, 数据之间的分列符用`&`表示

`\cline{i-j}`表示从 i 列的左侧到第 j 列起画一条水平线

2.2 跨行表格宏包 multirow

LaTeX 是一行一行排版表格的, 横向合并单元格较为容易, 由 `\multicolumn` 命令实现:

```
1 \multicolumn{n}{column-spec}{item}
```

其中 n 为要合并的列数, *column-spec* 为合并单元格后的列格式, 只允许出现一个 *l/c/r* 或 *p* 格式。如果合并前的单元格前后带表格线 |, 合并后的列格式也要带 | 以使得表格的竖线一致.item 则是表格的内容。

```
\begin{tabular}{|c|c|c|}
\hline
1 & 2 & Center \\ \hline
\multicolumn{2}{|c|}{3} & \\
\multicolumn{1}{r|}{Right} & \\ \hline
4 & \multicolumn{2}{|c|}{C} \\ \hline
\end{tabular}
```

1	2	Center
3		Right
4	C	



笔记 该命令可以用来修改某一个单元格的列格式。`\multicolumn{1}{column-spec}{item}`

纵向合并单元格需要用到 multirow 宏包提供的 `\multirow` 命令:

```
1 \multirow{所跨行数}{数据宽度}{数据}
```

```
\begin{tabular}{|c|c|}
\hline
\multirow{4}{*}{相亲数} & 220 \\
& 284 \\
& 1210 \\
& 1184 \\ \hline
\end{tabular}
```

相亲数	220
	284
	1210
	1184

我们看一个结合 `\cline`、`\multicolumn` 和 `\multirow` 命令的例子:

```
\begin{tabular}{ccc}
\hline
\multirow{2}{*}{Item} & \multicolumn{2}{c}{Value} \\
\cline{2-3}
& First & Second \\
\hline
A & 1 & 2 \\ \hline
\end{tabular}
```

Item	Value	
	First	Second
A	1	2

2.3 表格的浮动环境

控制表格在本页显示位置需要 table 环境, 参数有 htpb 四种, 分别表示 here, top, bottom, page of float.

table 环境中可以通过 \caption 命令为表格添加标题.

2.3.1 位置参数

1. *h*: 指定将该浮动体就地放置, 即放置在该浮动环境所在的位置, 如果版面所剩空间放不下该浮动体, 作者又没有指定其他选项, 系统则将 *h* 改为 *t*, 即尝试将该浮动体放置到下一页的顶部
2. *t* 表示将该浮动体放置到当前页或下一页的顶部
3. *b* 表示将该浮动体放置到当前页或下一页的底部
4. *p* 表示将该浮动体放置在当前页之后的单独一页, 该页被称为浮动体页 page of float.

该命令的使用方法如下:

```

1 \begin{table}[位置参数]
2 \caption{表格标题}
3 \begin{tabular}[<垂直对齐方式>]{<列格式说明>}
4 <单元格内容> & ... & <单元格内容> \\
5 .....
6 <单元格内容> & ... & <单元格内容> \\
7 \end{tabular}
8 \end{table}

```



笔记 在使用过程中 htpb 浮动环境经常会出现各种问题, 这时候我个人一般会使用 float 宏包, 将浮动参数改为 H, 从而强制将其放置在当前位置, 但是据某些大佬说, 这样的命令有点不合理。但是用起来是真的爽!

2.3.2 示例

表 2.1: 表格的标题

< 单元格 >	< 单元格内容 >	< 单元格 >
< 单元格 >	< 单元格内容 >	< 单元格 >

这是对代码

```

1 \begin{table}[H]
2 \centering %表格居中
3 \caption{表格的标题}
4 \begin{tabular}[c]{|l|c|r|} \hline
5 <单元格> & <单元格内容> & <单元格> \\ \hline
6 <单元格> & <单元格内容> & <单元格> \\ \hline
7 \end{tabular}
8 \end{table}

```

2.4 三线表

首先导入表格线宏包 booktabs 用于画表格线，常见的表格线有这么几个

- `\toprule` 画表格顶部的粗线
- `\midrule` 画表格中部的细线
- `\bottomrule` 画表格底部的粗线
- `\cmidrule{a-b}` 和 `\cline`相似

表 2.2: 这是成绩表

姓名	姓名 2	语文	语文 2	数学	英语
			90	80	70
			80	70	90
	小明		90	80	70
	小米		80	70	90

这是对代码

```

1 \begin{table}[H]
2 \centering
3 \caption{这是成绩表}
4 \begin{tabular}m{3cm}<{\centering}m{3cm}<{\centering}m{2cm}<{\centering}m{2cm}}
5 \toprule
6 姓名\qqquad 姓名2 & 语文\quad 语文2 & 数 学 & 英 语 \\ \midrule
7 \multirow{2}{*}{~} & 90 & 80 & 70 \\ \cmidrule{2-4}
8 ~ & 80 & 70 & 90 \\ \hline
9 小明 & 90 & 80 & 70 \\ \hline
10 小米 & 80 & 70 & 90 \\
11 \bottomrule
12 \end{tabular}
13 \end{table}

```

2.5 跨页长表格

顾名思义这是因为有时候表格太长，导致该页放不下，从而需要对表格进行跨页处理！

表 2.3: Python 数据分析与应用课程目录

课序号	课时名称
	第 1 模块: Python 数据分析概述
课时 1	1.1.1 掌握数据分析的基本概念 07:16
课时 2	1.1.2 了解数据分析应用场景 13:24
课时 3	1.2.1 了解 Python 数据分析的优势 02:19
课时 4	1.2.2 了解 Python 数据分析常用类库 08:11
课时 5	1.3 安装 Python 的 Anaconda 发行版 07:35
课时 6	1.4 掌握 Jupyter Notebook 常用功能 13:14

	第 2 模块: NumPy 数值计算基础
课时 7	2.1.1 创建数组对象 11:50
课时 8	2.1.2 生成随机数 01:52
课时 9	2.1.3 通过索引访问数组 03:49
课时 10	2.1.4 交换数组的形态 03:49
课时 11	2.2.1 创建 NumPy 矩阵 04:06
课时 12	2.2.2 掌握 ufunc 函数 04:31
课时 13	2.3.1 读写文件 06:50
课时 14	2.3.2 使用函数进行简单的统计分析 06:42
	第 3 模块: Matplotlib 数据可视化基础
课时 15	3.1.1 掌握 Python 基础语法
课时 16	3.1.2 设置 Python 的动态 rc 参数 05:32
课时 17	3.2.1 绘制散点图 07:10
课时 18	3.2.2 绘制折线图 03:21
课时 19	3.2.3 任务实现 01:42
课时 20	3.3.1 绘制直方图 05:00
课时 21	3.3.2 绘制饼图 03:28
课时 22	3.3.3 绘制箱线图 04:00
课时 23	3.3.4 任务实现 04:02
课时 24	3.4 小结 00:44

示例代码:

```

1 \begin{longtable}{|m{3cm}<{\centering}|m{11cm}|}
2 \caption{Python数据分析与应用课程目录} \\\hline
3 课序号 & 课时名称 \\\hline
4 ~ & 第1模块: Python数据分析概述 \\\hline
5 课时1 & 1.1.1 掌握数据分析的基本概念 07:16 \\\hline
6 课时2 & 1.1.2 了解数据分析应用场景 13:24 \\\hline
7 课时3 & 1.2.1 了解Python数据分析的优势 02:19 \\\hline
8 课时4 & 1.2.2 了解Python数据分析常用类库 08:11 \\\hline
9 课时5 & 1.3 安装Python的Anaconda发行版 07:35 \\\hline
10 课时6 & 1.4 掌握Jupyter Notebook常用功能 13:14 \\\hline
11 ~ & 第2模块: NumPy数值计算基础 \\\hline
12 课时7 & 2.1.1 创建数组对象 11:50 \\\hline
13 课时8 & 2.1.2 生成随机数 01:52 \\\hline
14 课时9 & 2.1.3 通过索引访问数组 03:49 \\\hline
15 课时10 & 2.1.4 交换数组的形态 03:49 \\\hline
16 课时11 & 2.2.1 创建NumPy矩阵 04:06 \\\hline
17 课时12 & 2.2.2 掌握ufunc函数 04:31 \\\hline
18 课时13 & 2.3.1 读写文件 06:50 \\\hline
19 课时14 & 2.3.2 使用函数进行简单的统计分析 06:42 \\\hline
20 ~ & 第3模块: Matplotlib数据可视化基础 \\\hline
21 课时15 & 3.1.1 掌握Python基础语法 08:54 \\\hline
22 课时16 & 3.1.2 设置Python的动态rc参数 05:32 \\\hline

```

```

23 课时17 & 3.2.1 绘制散点图 07:10 \\ \hline
24 课时18 & 3.2.2 绘制折线图 03:21 \\ \hline
25 课时19 & 3.2.3 任务实现 01:42 \\ \hline
26 课时20 & 3.3.1 绘制直方图 05:00 \\ \hline
27 课时21 & 3.3.2 绘制饼图 03:28 \\ \hline
28 课时22 & 3.3.3 绘制箱线图 04:00 \\ \hline
29 课时23 & 3.3.4 任务实现 04:02 \\ \hline
30 课时24 & 3.4 小结 00:44 \\ \hline
31 \end{longtable}

```



笔记 跟其他表格其实没什么太大区别，换个环境而已！

2.6 其他宏包与表格缩放

还有些对角线宏包，就是有时候需要在表格左上角的单元格画一条对角线，线上是列标题，线下是行标题，这时候可以使用 `slashbox` 宏包，不知道这玩意还更新没，一般很少使用！还有些彩色表格宏包 `colorbl`，一般很少用，可调列宽的表格宏包 `tabulax`，在 `tabular*` 环境中可以设置表格的总宽，但是其只是增减列间空白，使表格本身伸展到设定宽度，而无法控制数据列的宽度，可能会造成有些数据凸出表格。放两个例子感受一下。

<pre> \begin{tabular*}{50mm}{ll}\hline 年份 & 探测目标 \\ \hline 1959 & 苏联月球3号发回月球背面照片。 \\ 1964 & 美国水手4号飞往火星。 \\ \hline \end{tabular*} </pre>	<table> <tr> <th>年份</th><th>探测目标</th></tr> <tr> <td>1959</td><td>苏联月球 3 号发回月球背面照片。</td></tr> <tr> <td>1964</td><td>美国水手 4 号飞往火星。</td></tr> </table>	年份	探测目标	1959	苏联月球 3 号发回月球背面照片。	1964	美国水手 4 号飞往火星。
年份	探测目标						
1959	苏联月球 3 号发回月球背面照片。						
1964	美国水手 4 号飞往火星。						
<pre> \begin{tabularx}{50mm}{lX}\hline 年份 & 探测目标 \\ \hline 1959 & 苏联月球3号发回月球背面照片。 \\ 1964 & 美国水手4号飞往火星。 \\ \hline \end{tabularx} </pre>	<table> <tr> <th>年份</th><th>探测目标</th></tr> <tr> <td>1959</td><td>苏联月球 3 号发回月球背面照片。</td></tr> <tr> <td>1964</td><td>美国水手 4 号飞往火星。</td></tr> </table>	年份	探测目标	1959	苏联月球 3 号发回月球背面照片。	1964	美国水手 4 号飞往火星。
年份	探测目标						
1959	苏联月球 3 号发回月球背面照片。						
1964	美国水手 4 号飞往火星。						



笔记 后面的这些我个人觉得用的比较少，常用的就是普通表格，三线表和长表格了！这是应该重点掌握的！

有时候希望适当调整表格的整体外形尺寸，以获得最佳的排版效果，虽然改变字体可以改变表格整体尺寸，但是很难控制外形。这时候可以在导言区调用插图宏包 `graphicx` 然后将需要缩放的表格置于该缩放命令当中：

```


1 \begin{table}[H]
2 \centering
3 \caption{这是成绩表}
4 \scalebox{0.9}[1.25]{
5 \begin{tabular}{m{3cm}<{\centering}m{3cm}<{\centering}m{2cm}<{\centering}m{2cm}}
6 \toprule
7 姓名\qqquad 姓名2 & 语文\quad 语文2 & 数学 & 英语 \\ \midrule
8 \multirow{2}{*{~}} & 90 & 80 & 70 \\ \cmidrule{2-4}

```

```
9 ~ & 80 & 70 &90 \\ \hline
10 小明 & 90 &80 &70 \\ \hline
11 小米 & 80 & 70 &90 \\
12 \bottomrule
13 \end{tabular}}
14 \end{table}
```

表 2.4: 这是成绩表

姓名	姓名 2	语文	语文 2	数学	英语
		90		80	70
		80		70	90
小明		90		80	70
小米		80		70	90

 **笔记** 有时可以用到，做表格调整时！

2.7 列表的使用

列表呢无非就是两种，有序列表和无序列表，区别一个有序号一个没序号了！

2.7.1 无序列表

召唤一个无序列表如下：


```
\begin{itemize}
\item 这是无序1
\item 这是无序2
\item [*] 这是无序3,自定义序号样式
\end{itemize}
```

- 这是无序 1
- 这是无序 2
- * 这是无序 3, 自定义序号样式

一个比较好玩的列表环境 dinglist

```
\begin{dinglist}{47}
\item 这是无序1
\item 这是无序2
\item 这是无序3
\end{dinglist}
```

- ☞ 这是无序 1
- ☞ 这是无序 2
- ☞ 这是无序 3

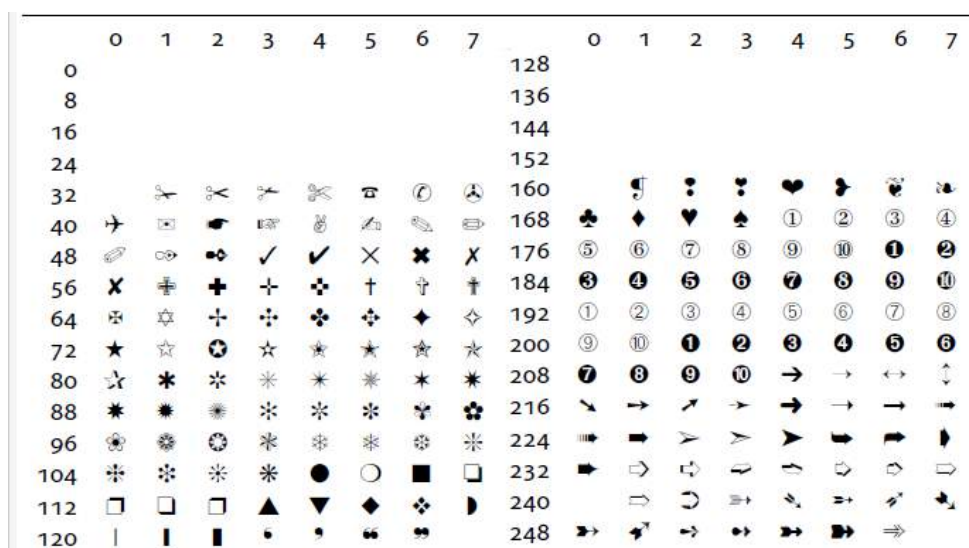
 **笔记** dinglist 这个列表环境参照了 pifont 这个宏包，它里面有 200 多个符号，通过更改数字进行替换，使用时需要导入 pifont 这个宏包，对了这些还可以进行嵌套。


这里有个示例可以参照下：



<code>\begin{dinglist}{47}</code>	
<code>\item 这是无序1</code>	⇒ 这是无序 1
<code>\begin{dinglist}{118}</code>	
<code>\item 这是无序1.1</code>	❖ 这是无序 1.1
<code>\item 这是无序1.2</code>	❖ 这是无序 1.2
<code>\item 这是无序1.3</code>	❖ 这是无序 1.3
<code>\end{dinglist}</code>	
<code>\item 这是无序2</code>	⇒ 这是无序 2
<code>\begin{dinglist}{227}</code>	
<code>\item 这是无序2.1</code>	➤ 这是无序 2.1
<code>\item 这是无序2.2</code>	➤ 这是无序 2.2
<code>\item 这是无序2.3</code>	➤ 这是无序 2.3
<code>\end{dinglist}</code>	
<code>\item 这是无序3</code>	⇒ 这是无序 3
<code>\begin{dinglist}{118}</code>	
<code>\item 这是无序3.1</code>	❖ 这是无序 3.1
<code>\item 这是无序3.2</code>	❖ 这是无序 3.2
<code>\item 这是无序3.3</code>	❖ 这是无序 3.3
<code>\end{dinglist}</code>	
<code>\end{dinglist}</code>	

这里把 pifont 这个宏包的符号截图给出来：



 **笔记** 凑合看吧，懒得分图了！，有序差不多就是这些内容！基本上够用了。

2.7.2 有序列表

召唤一个有序列表：

<pre>\begin{enumerate} \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>1. 这是有序 1 2. 这是有序 2 3. 这是有序 3</pre>
--	--



笔记 关于有序列表是不是可以自定义序号的样式，答案是肯定有，但是涉及到修改命令了，有点费事，就不说了，空了可以自己去看，这里给出一些简单方法实现。

<pre>\begin{enumerate}[(I)] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>(I) 这是有序 1 (II) 这是有序 2 (III) 这是有序 3</pre>
---	--

<pre>\begin{enumerate}[(i)] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>(i) 这是有序 1 (ii) 这是有序 2 (iii) 这是有序 3</pre>
---	--

<pre>\begin{enumerate}[(A)] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>(A) 这是有序 1 (B) 这是有序 2 (C) 这是有序 3</pre>
---	---

<pre>\begin{enumerate}[(a)] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>(a) 这是有序 1 (b) 这是有序 2 (c) 这是有序 3</pre>
---	---

<pre>\begin{enumerate}[{a}-1] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>a-1 这是有序 1 a-2 这是有序 2 a-3 这是有序 3</pre>
---	---

<pre>\begin{enumerate}[Ex i] \item 这是有序1 \item 这是有序2 \item 这是有序3 \end{enumerate}</pre>	<pre>Ex i 这是有序 1 Ex ii 这是有序 2 Ex iii 这是有序 3</pre>
--	---



笔记 有序的嵌套，其实和无序的嵌套是一样的，，符号也是可以自定义的，可以看看简单的嵌套

```

\begin{enumerate}
\item 这是有序1
\begin{enumerate}
\item 这是有序1.1
\item 这是有序1.2
\item 这是有序1.3
\end{enumerate}
\item 这是有序2
\begin{enumerate}
\item 这是有序2.1
\item 这是有序2.2
\item 这是有序2.3
\end{enumerate}
\item 这是有序3
\begin{enumerate}
\item 这是有序3.1
\item 这是有序3.2
\item 这是有序3.3
\end{enumerate}
\end{enumerate}

```

1. 这是有序 1
 - (a). 这是有序 1.1
 - (b). 这是有序 1.2
 - (c). 这是有序 1.3
2. 这是有序 2
 - (a). 这是有序 2.1
 - (b). 这是有序 2.2
 - (c). 这是有序 2.3
3. 这是有序 3
 - (a). 这是有序 3.1
 - (b). 这是有序 3.2
 - (c). 这是有序 3.3



笔记 其实我在做这块的时候, 更想要的是那种带圈的数字, 如何实现的, 复杂方法的我们可以通过改命令实现, 简单的我们可以掉包实现 dingautolist, 注意这里貌似只有 172 和 202 才可以用, 其他的不行, 最大排到 10, 想更多的话, 改命令实现!!!

```

\begin{dingautolist}{172}
\item 这是有序1
\item 这是有序2
\item 这是有序3
\end{dingautolist}

```

- ① 这是有序 1
- ② 这是有序 2
- ③ 这是有序 3

```

\begin{dingautolist}{202}
\item 这是有序1
\item 这是有序2
\item 这是有序3
\end{dingautolist}

```

- ❶ 这是有序 1
- ❷ 这是有序 2
- ❸ 这是有序 3

2.7.3 解说列表

不废话这玩意就是对一些专业术语做解释用的, 直接看示例:

```

\begin{description}
\item[词条1] 这是词条的解释语言.....
\item[词条2] 这是词条的解释语言.....
\item[词条3] 这是词条的解释语言.....
\end{description}

```

词条 1 这是词条的解释语言.....
 词条 2 这是词条的解释语言.....
 词条 3 这是词条的解释语言.....

2.7.4 通用列表环境 list

```

1 \begin{list}{默认编号}{声明}
2   \item [标号] 条目1
3   \item [标号] 条目2
4   .....
5   \item [标号] 条目?
6 \end{list}

```

默认标号 在条目之前加入的标号，若条目命令\item没有给出可选参数标号的话，就使用默认标号了，都不设置，那就都没了

声明 针对条目标号，字体，尺寸等样式的设置命令



笔记 不会高级用法，那就针对每个条目手动编号，没办法技术达不到！看下示例

```

\begin{list}{}{}
\item [A-I] 这是条目1
\item [A-II] 这是条目2
\end{list}

```

A-I 这是条目 1

A-II 这是条目 2

```

\begin{list}{}{}
\item [{\bfseries 步骤1:}] 这是条目1
\item [{\bfseries 步骤2:}] 这是条目2
\end{list}

```

步骤 1: 这是条目 1

步骤 2: 这是条目 2

2.7.5 列表宏包 enumerate

这玩意看看就行，别较真！

```

\begin{enumerate}[label=\fcolorbox{blue!30}
{cyan!40}{\ding{\value*}},start=172
,itemsep=0pt]
\item 这是步骤1
\item 这是步骤2
\item 这是步骤3
\end{enumerate}

```

① 这是步骤 1

② 这是步骤 2

③ 这是步骤 3

第三章 LaTeX 入门之插图篇

内容提要

- 图形分类
- 基本插图
- 插图参数
- 图片标题与浮动
- 多个插图放置
- eps 图像获取

3.1 图形的分类

3.1.1 位图图像

也成为点阵图形，技术上称为栅格图像，它使用称作像素点的小方形组成的网格表示图像，每个像素有自己特定的位置和颜色值。

位图图像可分为无损压缩格式和有损压缩格式。无损压缩格式的有点事能够完整的保存图像的像素信息，但是压缩率比较低。有损压缩技术可以大幅度压缩图形文件，但是会使图形的像素降低。常用的图形文件中 TIFF、PNG 和 GIF 都是无损压缩格式，jpg 是有损压缩格式。

3.1.2 向量图形

向量图形是由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、长短和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，可以将其任意缩放旋转，按任意分辨率打印都不会失真。因此这是最适合的图形，但是获取难度较大。

3.1.3 插图的基本命令

当需要在源文件插入图片时，首先应当调用 *David Carlisle* 编写的 `graphicx` 宏包。使用方法如下：

```
1 \usepackage{graphicx} %将该命令放置在导言区
```

然后使用插图命令：

```
1 \includegraphics[参数1=选项,参数2=选项,...]{插图}
```



笔记 插图的参数通常很多，下面我们会介绍常用的一些插图参数！召唤一个图片感受下先。

```
\includegraphics[width=1\linewidth]{welt.jpg}
```





笔记 这里需要注意的是,插图和图片要和你的编译的源文件放在同一目录下,上述[width=1\linewidth]这个参数就是说,插图的宽度等于当前文本行的宽度,

下面我们会具体介绍插图的一些参数.

3.2 插图命令的参数

在 graphicx 宏包中提供的各种与插图有关的命令中最常用的就是插图命令,即就是上面写到的这个命令:

```
\includegraphics[参数1=选项,参数2=选项,...]{插图}
```

其中插图是所要插入图形的名称,包括扩展名,下来就是这些参数,大概可被分为三类,外形参数,裁剪参数,和布尔参数。其中我们平时写作中用的是最多的是外形参数,关于裁剪参数,其中的命令主要用于设置插图的显示区域,一般很少用到,因为在插图时一般会直接将想要插入的图片裁减好才进行插入,当然你想在插入的时候裁剪,你可以自己找下参考书,或者找度娘。关于布尔参数,用的也比较少,一般我们在插图过程中设定的高度或者宽度不成比例的情况下,可能会造成图片失真,这时候通过设置布尔参数中的某些参数,将会按照原图的高宽比例缩放放到设定的宽度或高度,但不会超出所设定高度或者宽度。



笔记 关于三种参数,我们必须掌握的是外形参数,知道这些参数足以解决百分之九十的问题。

3.2.1 外形参数说明

- ❖ height: 设定插图的高度,可使用系统认可的长度单位
- ❖ totalheight: 设置插图的总高度。总高度 = 高度 + 深度。
- ❖ width: 设置插图的宽度。
- ❖ scale: 设置图片的缩放系数, scale=2, 表示将图片放大两倍插入,当它是个负值时,则表示在缩放的同时,将插图逆时针旋转 180°
- ❖ 还有个 origin 用于设置插图的旋转点,很少用,bb 参数用于设定 EPS 格式图片的 BoundingBox 值,一个插图的坐标值。

3.2.2 关于图片的高度与深度

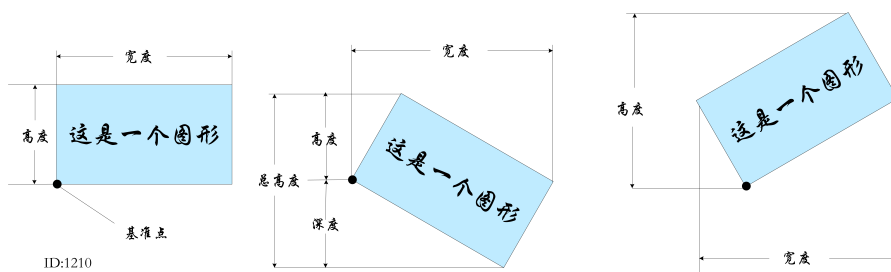


图 3.1: 这是图片的高度与深度

关于高度和总高度这些我个人一般很少去定义,因为直接这样定义图片可能会变得很难看,通常情况都是通过设置宽度和当前文本行的数值关系去设置,高度这时会自动调整,或者使用缩放这个命令进行调整。具体示例如下:

设置图像宽度是文本宽度的50%:

```
\includegraphics[width=0.5\linewidth]{welt.jpg}
```

设置图像宽度是文本宽度的 50%:



设置图像宽度和文本宽度相同:

```
\includegraphics[width=1\linewidth]{welt.jpg}
```

设置图像宽度和文本宽度相同:



3.2.3 设置图片的缩放

图片的缩放可以这样子去设置。

这里的图片缩放系数为原图的1/10:

```
\includegraphics[scale=0.1]{12101.jpg}
```

这里的图片缩放系数为原图的 1/10:



这里的图片缩放系数为原图的1/20:

```
\includegraphics[scale=0.05]{12101.jpg}
```

这里的图片缩放系数为原图的 1/20:



笔记 关于负值的设置, 感觉没必要, 可以先设置缩放, 然后再设置旋转, 直接使用负值进行缩放旋转, 图片样式不太好看, 这里不做展示, 有兴趣可以试试:

```
1
2 \includegraphics[scale=0.1]{12101.jpg}
3
4 \includegraphics[scale=-0.1]{12101.jpg}
```

3.2.4 关于图片的旋转

图片的旋转一般情况用的很少, 但是偶尔也会使用。

<p>默认不旋转</p> <pre>\includegraphics[width=0.6\linewidth]{121.png}</pre>	<p>默认不旋转</p> 
<p>旋转180°,默认旋转点为1B基准点</p> <pre>\includegraphics[width=1\linewidth,angle=180]{121.png}</pre>	<p>旋转 180°, 默认旋转点为 1B 基准点</p> 

这里给出旋转中心的参考图，可以根据旋转点的设置，从而实现不同的旋转效果！就不再一一介绍了。

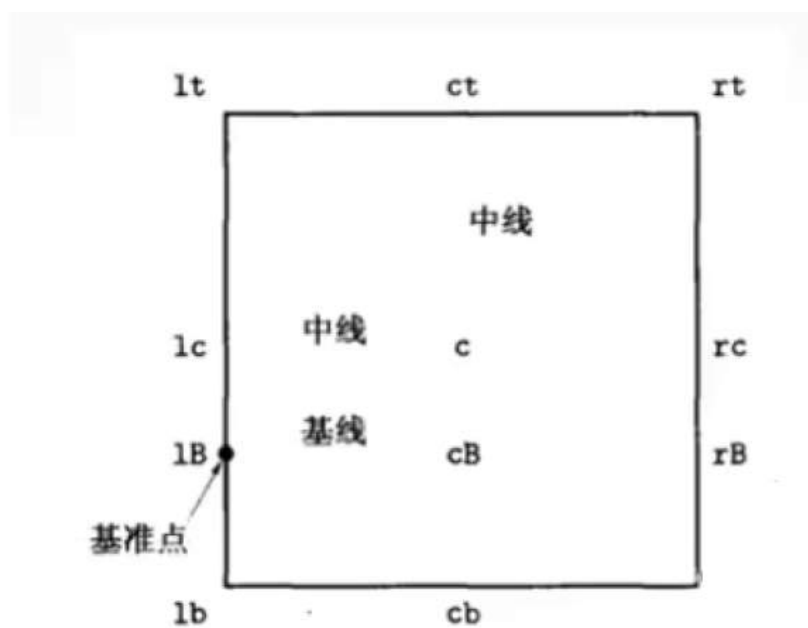


图 3.2: 旋转中心参考

3.3 图片的标题和浮动

在以上内容中我们只是简单的插入一张图片，并未做图片标题和图片位置的设置，这时不合理的，现在我们看下如何设置这些内容，同表格一样，表格有一个 `table` 环境用于设置这些信息，图片的话，我们则可以使用 `figure` 环境进行设置。示例如下：


```

\begin{figure}[H]
\centering
\includegraphics[width=0.8\linewidth]
{figure/welt}
\caption{这是薇尔莉特}
\end{figure}

```



图 3.3: 这是薇尔莉特

关于命令的解释如下：

1. `\centering`这是用来使图片居中用
2. `\caption{标题}`这是用来设置标题信息的
3. `\begin{figure}[H]`这里给出了一个 H 参数，设置浮动用的



笔记 关于浮动，默认情况是 `htbp` 参数，具体可以参照表格浮动部分，这里就不在赘述了。两者基本是一样的

3.4 多个插图放置

这个才是关心的重点!!

3.4.1 两图并列，共享标题

```

\begin{figure}[H]
\centering
\includegraphics[width=0.45\linewidth]{22.jpg}
%插入的第一个图片
\includegraphics[width=0.45\linewidth]{22.jpg}
%插入的第二张图片
\caption{这是日向}
\end{figure}

```

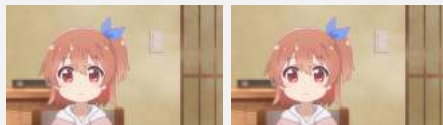


图 3.4: 这是日向

貌似这样子有点小了！后面用代码框吧！



图 3.5: 紫罗兰永恒花园

1 `\begin{figure}[H]`

```

2 \centering
3 \includegraphics[width=0.45\linewidth]{welt1.jpg}
4 %插入的第一个图片
5 \includegraphics[width=0.45\linewidth]{welt.jpg}
6 %插入的第二张图片
7 \caption{紫罗兰永恒花园}
8 \end{figure}

```


 **笔记** 这里其实容易出问题的是图片大小不一样怎么版，通常我的解决办法是，从一开始就把两图片的大小设置成一样的，或者通过改参数实现。关于参数如何改，多改改就用经验了！O(_)O



图 3.6: 紫罗兰永恒花园

```

1 \begin{figure}[H]
2 \centering
3 \includegraphics[width=0.4\linewidth]{welt1.jpg}
4 %插入的第一个图片
5 \includegraphics[width=0.53\linewidth]{welt.jpg}
6 %插入的第二张图片
7 \caption{紫罗兰永恒花园}
8 \end{figure}

```

3.4.2 两图并列，各有标题，共享标题

3.4.2.1 实现方法 1

一图胜千言，直接看效果！



图 3.7: 白骑士·月光



图 3.8: 御神装·勿忘

图 3.9: 这是崩坏三

实现代码：

```

1 \begin{figure}[H]
2 \begin{minipage}{0.5\linewidth}
3 \centering
4 \includegraphics[width=0.9\linewidth]{yg.jpg}
5 \caption{白骑士·月光}
6 \end{minipage}
7 \begin{minipage}{0.5\linewidth}
8 \centering
9 \includegraphics[width=0.8\linewidth]{12101.jpg}
10 \caption{御神装·勿忘}
11 \end{minipage}
12 \caption{这是崩坏三}
13 \end{figure}

```

代码解释：

这里使用了一个 minipage 环境，第一个 minipage 设置环境的长度等于 1/2 的线宽，第二个环境也是一样的，然后外面套一个大的 figure 环境就可以了！

3.4.2.2 实现方法 2



(a) 白骑士·月光



(b) 御神装·勿忘

图 3.10: 这是崩坏三

实现代码：

```

1 \begin{figure}[H]
2 \centering
3 \subfloat[白骑士·月光]{
4 \includegraphics[width=0.48\linewidth]{yg.jpg}}
5 \hspace{10pt}
6 \subfloat[御神装·勿忘]{
7 \includegraphics[width=0.4\linewidth]{12101.jpg}}
8 \caption{这是崩坏三}
9 \end{figure}

```

代码解释：

首先你需要在导言区导入 subfig 浮动体宏包，然后就是关于它的用法：

```

1 %导言区

```

```

2 \usepackage{subfig}
3 %正文区
4 \subfloat[表格标题]]{图形或者表格}

```

然后两者嵌套即可，其中`\hspace{10pt}`这是为了设置水平间距用的！

3.4.2.3 实现方法 3

还可以通过前面学过的表格环境进行实现



(白骑士·月光)



(御神装·勿忘)

图 3.11: 这是崩坏三

代码实现：

```

1 \begin{figure}[H]
2 \centering
3 \begin{tabular}{cc}
4 \includegraphics[width=0.5\linewidth]{yg.jpg} &
5 \includegraphics[width=0.42\linewidth]{12101.jpg} \\
6 (白骑士·月光) & (御神装·勿忘) \\
7 \end{tabular}
8 \caption{这是崩坏三}
9 \end{figure}

```

3.4.3 其他插图

figure+subfloat



(a) pic1



(b) pic2



(c) pic3

图 3.12: 这是总标题

代码实现：



```

1 \begin{figure}[H]
2 \centering
3 \subfloat[pic1]{\label{pic1}}
4 \includegraphics[width=0.3\linewidth]{1.png}}
5 \hspace{10pt}
6 \subfloat[pic2]{\label{pic1}}
7 \includegraphics[width=0.3\linewidth]{2.png}}
8 \subfloat[pic3]{\label{pic1}}
9 \includegraphics[width=0.3\linewidth]{3.png}}
10 \caption{这是总标题}
11 \end{figure}

```

figure+tabular



(a)



(b)



(c)



(d)

图 3.13: 四个日向

代码实现

```

1 \begin{figure}[H]
2 \begin{tabular}{cc}
3 \includegraphics[width=0.4\linewidth]{121.png} & \\
4 \includegraphics[width=0.4\linewidth]{121.png} & \\
5 (a) & (b) \\
6 \includegraphics[width=0.4\linewidth]{121.png} & \\
7 \includegraphics[width=0.4\linewidth]{121.png} & \\
8 (c) & (d) \\
9 \end{tabular}
10 \caption{四个日向}
11 \vspace{-0.5em}
12 \end{figure}

```


figure+minipage



图 3.14: 这是图一



图 3.15: 这是图一



图 3.16: 这是图三



图 3.17: 这是图四

图 3.18: 这是总的标题

代码实现

```

1 \begin{figure}[H]
2 \begin{minipage}{0.5\linewidth}
3 \centering
4 \includegraphics[width=0.9\linewidth]{111.png}
5 \caption{这是图一}
6 \end{minipage}
7 \begin{minipage}{0.5\linewidth}
8 \centering
9 \includegraphics[width=0.9\linewidth]{111.png}
10 \caption{这是图一}
11 \end{minipage}
12
13 \begin{minipage}{0.5\linewidth}
14 \centering
15 \includegraphics[width=0.9\linewidth]{111.png}
16 \caption{这是图三}
17 \end{minipage}
18 \begin{minipage}{0.5\linewidth}
19 \centering
20 \includegraphics[width=0.9\linewidth]{111.png}
21 \caption{这是图四}
22 \end{minipage}
23
24 \caption{这是总的标题}
25 \end{figure}

```



笔记 minipage 是采用行方向累加制的自动排版方法,也就是当子图的 width 累计大于等于\linewidth时,后续的图片自动排到下一行,关于这里面的区别,表格应该是最简单好理解的,但是问题是,表格实现的不能进行引用,而使用 minipage 或者其他实现的可以进行交叉引用,关于交叉引用看吧,空了在参考文献部分讲吧!那玩意不是太难!

3.5 eps 图像获取

```
1 for /f %%i in ('dir /b *.jpg *.jpeg *.bmp *.png') do (  
2 @echo %%i  
3 bmeps -c %%i %%~ni.eps  
4 @echo Finished  
5 )  
6 pause
```

第四章 LaTeX 入门之参考文献篇

内容提要

❑ 参考文献

❑ 代码框设计

❑ 交叉引用

❑ 网址链接

这一部分本来只想写参考文献的，但是还是打算把其他内容补充进去，就是这样子！

4.1 参考文献

这部分主要介绍两种参考文献的使用方式，一种是基本的参考文献和引用。另外一种则是使用 bibtex 数据库进行参考文献的管理。

4.1.1 什么是参考文献

参考文献是在学术研究过程中，对某一著作或论文的整体参考或借鉴。征引过的文献在注释中已注明，不再出现于文后参考文献中。

按照字面的意思，参考文献是文章或著作等写作过程中参考过的文献。然而，按照 GB/T 7714-2015《信息与文献参考文献著录规则》的定义，文后参考文献是指：“为撰写或编辑论文和著作而引用的有关文献信息资源。根据《中国学术期刊（光盘版）检索与评价数据规范（试行）》和《中国高等学校社会科学学报编排规范（修订版）》的要求，很多刊物对参考文献和注释作出区分，将注释规定为“对正文中某一内容作进一步解释或补充说明的文字”，列于文末并与参考文献分列或置于当页脚地。

4.1.2 基本的参考文献

首先我们召唤一波参考文献！

- 1 刘国钧，陈绍业，王凤翥. 图书馆目录[M]. 北京：高等教育出版社，1957.15-18.
- 2
- 3 辛希孟. 信息技术和信息服务国际研讨会论文集：A集[C]. 北京：中国社会科学出版社，1994.
- 4
- 5 张筑生. 微分半动力系统的不变集[D]. 北京：北京大学数学系数学研究所，1983.
- 6
- 7 冯西桥. 核反应堆压力管道和压力容器的LBB分析[R]. 北京：清华大学核能技术设计研究院，1997.

这是随便找的一些参考文献，两步问题，首先这玩意怎么插到文章，然后如何引用这些参考文献。关于参考文献的书写格式，因为各个期刊要求的都不太相同，所以没办法统一介绍，这部分就需要你参照你写文章的期刊是怎么要求了。

现在我们先看下参考文献的环境是什么样的。如下代码所示：

```
1 \begin{thebibliography}{最大序号}
2 \bibitem[文献序号1]{检索名} 文献信息
3 \bibitem[文献序号2]{检索名} 文献信息
```



```

4 \bibitem[文献序号3]{检索名} 文献信息
5 .....
6 \end{thebibliography}

```

参数说明如下：

1. 最大序号：用于测定文献列表中文献序号的最大宽度，如果你是 10 以内的参考文献，那就用 9，超过 10 小于 100 那就填 99。
2. 文献序号：可选参数，用于设定该条文献在参考文献列表中的序号。
3. 检索名：为该文献信息起的简短名称，
4. 文献信息：就是参考文献内容了。

这就是参考文献的书写了，然后就是关于参考文献的引用。

4.1.3 参考文献的引用

如果要在正文中引用参考文献列表中的文献时，可以在引文之后插入文献的引用命令：

```

1 \cite[附加信息]{检索名1,检索名2....}

```

参数说明如下：

1. 检索名：就是文献条目中的\bibitem中的检索名，引用那条文献，就指定那条文献的检索名，可以指定多个检索名，中间使用逗号分割（半角逗号）且不留空格。
2. 附加信息：可选参数，可以用作对所引用参考文献的注解。例如文献过长，可以使用附加信息说明参考内容的页码范围。

4.1.4 引文格式修改

还是比较有用的吧！在引用的标志中如果有多个参考文献序号，可能会出现 [4,6,5] 这种跳号情况，文献序号之间间隔无法调整，这时候可以调用 cite 宏包进行修改。使用命令如下：

```

1 \usepackage[格式]{cite}

```

命令说明，可以通过修改格式这个可选参数来对引用格式进行修改，常见参数如下：

1. biblable: 将参考文献的列表中的文献序号改成上标形式
2. noadjust: 不在引文与\cite中插入一个空格，默认是 adjust，会插入一个空格。
3. nocompress: 不将引用标志中三个以上的连续序号改为范围序号，差不多是这样，如果你默认是 [4, 5, 6] 那么系统默认会给你改成 [4 – 6]
4. nosort: 不对引用标志的序号进行排序，默认是排的。
5. nospace: 取消分隔逗号的空格。默认是有的。
6. ref: 在序号前加入参考文献缩写：Ref。
7. super: 取消引用标志，将其中的序号改用上标格式显示，这是引用命令中不能出现附加信息的可选参数，否则仍按默认选项

或者使用 natbib 宏包，看起来这个更好用些，natbib 重新实现了 \cite 命令以适应作者-年和编号两种形式的引用，完全兼容标准的文献样式 plain, alpha, unsrt 等，也可以配合 harvard, apalike, chicago, astron, authordate 等样式要求。

```

1 \usepackage[option]{natbib}

```

option 参数说明：

1. round: (default) 使用圆括号
2. square: 使用方括号
3. curly: 使用花括号
4. angle: 使用尖括号
5. colon: (default) 用引号分隔多个引用
6. comma: 用逗号分隔多个引用
7. authoryear: (default) 使用作者-年引用形式
8. numbers: 使用编号引用形式
9. super: 使用 Nature 那样的上标编号引用
10. sort: 多个引用按照首字母排序
11. sort&compress: 除排序外, 多个引用可以合并 (如 3-6, 15)
12. longnamesfirst: 多个作者的文献第一次被引用时列出所有作者, 以后的引用可以缩写为 et al.

注意有的文档在一开始就给你把参考文献的样式做过限定, 比如这个模板, 那么如何去更改呢? 在导言区重定义下你的命令就可以了, 解释涉及到命令修改, 算了!

```
\newcommand{\upcite}[1]{${\mbox{\scriptsize \cite{#1}}}$}
```

4.1.5 bibtex 文献管理使用

使用方式:

```
1 \bibliographystyle{文献格式名}
2 \bibliography{文献数据库名}
```

- 文献格式名, 这个一般有所选的参考期刊给定, 常用的 plain, unsrt, alpha, abbrev, ieetr 等等。
- 文献数据库, BibTeX 的 bib 文件是一个记录已阅文献的数据库, 但是通常不建议手动编译 bib 文件,

建议:

1. 使用 JabRef 或 Zotero 等文献管理工具导出 bib 文件创
2. 使用 Google Scholar 或 Bing 学术导出 bib 条目建

引文的信息还有很多国内外网站可以获取包括: 百度学术、搜狗学术, 万方、维普..... 很多,

有时候参考文献出现问题, 需要重构怎么办? 使用 notepad++ 等一些字处理软件进行更改, 但是不建议自己手动写 bib 文件。

4.2 交叉引用

使用/lable 标号来定义标号, 这里的标号可以是字母, 数字, 标点等组成的字符串. 需要引用, 则使用/ref 标号, 这里的” 标号” 应该是有/lable 定义过的, 定义和引用的先后无关.

4.3 网址链接

调用宏包:

```
1 \usepackage{hyperref}
```

```
1 \href{{文字}} 命令可以用来使文字产生指向URL地址的超链接效果。
```

4.4 代码框设计

```
1
2 \usepackage{listings}%插入代码
3
4 \lstset {numbers=left, %设置行号位置
5
6 numberstyle=\tiny, %设置行号大小
7
8 keywordstyle=\color{blue}, %设置关键字颜色
9
10 commentstyle=\color{cmyk}{1,0,1,0}, %设置注释颜色
11
12 frame=single, %设置边框格式
13
14 escapeinside="", %逃逸字符(1左面的键), 用于显示中文
15
16 %breaklines, %自动折行
17
18 extendedchars=false, %解决代码跨页时, 章节标题, 页眉等汉字不显示的问题
19
20 xleftmargin=0em,xrightmargin=0em, aboveskip=1em, %设置边距
21
22 tabsize=4, %设置tab空格数
23
24 %showspaces=false %不显示空格
25
26 breaklines,%自动换行
27
28 columns=flexible,%不随便添加空格,只在已经有空格的地方添加空格,
29
30 %如果想要添加空格使用fixed作为参数(这是默认的),如果坚决不添加空格使用fullflexible作为参数
```

这部分还有很多!

