

(25 条消息) Latex 各种命令、符号、公式、数学符号、排版等_latex \overset_CopperDong 的博客 - CSDN 博客

本文仅供学习参考使用，一切版权和解释权均归原作者所有，转载地址：<http://blog.csdn.net/garfield007/article/details/51646604>

数学符号详细内容见：<http://mohu.org/info/symbols/symbols.htm> 直接查看链接

<http://www.mohu.org/info/lshort-cn.pdf> 点直接查看链接

直接在 html 源文件中插入 ``

例如：

热力学第一定律可表述为 $Q + W = \Delta U$ ，其中 Q 是系统从环境获取的热量， W 是环境对系统做的功， ΔU 是系统内能的改变。

函数、符号及特殊字符

声调

语法	效果	语法	效果	语法	效果
<code>\bar{x}</code>	\bar{x}	<code>\acute{\eta}</code>	$\acute{\eta}$	<code>\check{\alpha}</code>	$\check{\alpha}$
<code>\grave{\eta}</code>	$\grave{\eta}$	<code>\breve{a}</code>	\breve{a}	<code>\ddot{y}</code>	\ddot{y}
<code>\dot{x}</code>	\dot{x}	<code>\hat{\alpha}</code>	$\hat{\alpha}$	<code>\tilde{\iota}</code>	$\tilde{\iota}$

函数

语法	效果	语法	效果	语法	效果
<code>\sin\theta</code>	$\sin\theta$	<code>\cos\theta</code>	$\cos\theta$	<code>\tan\theta</code>	$\tan\theta$
<code>\arcsin\frac{L}{r}</code>	$\arcsin\frac{L}{r}$	<code>\arccos\frac{T}{r}</code>	$\arccos\frac{T}{r}$	<code>\arctan\frac{L}{T}</code>	$\arctan\frac{L}{T}$
<code>\sinh g</code>	$\sinh g$	<code>\cosh h</code>	$\cosh h$	<code>\tanh i</code>	$\tanh i$
<code>\operatorname{sh}j</code>	$\operatorname{sh}j$	<code>\operatorname{argsh}k</code>	$\operatorname{argsh}k$	<code>\operatorname{ch}h</code>	$\operatorname{ch}h$

<code>\operatorname{argch}l</code>	$\operatorname{argch}l$	<code>\operatorname{th}i</code>	$\operatorname{th}i$	<code>\operatorname{argth}m</code>	$\operatorname{argth}m$
$k'(x)=\lim_{\Delta x\rightarrow 0}\frac{k(x)-k(x-\Delta x)}{\Delta x}$	$f'(x)=\lim_{\Delta x\rightarrow 0}\frac{f(x)-f(x-\Delta x)}{\Delta x}$	<code>\limsup S</code>	$\limsup S$	<code>\liminf I</code>	$\liminf I$
<code>\max H</code>	$\max H$	<code>\min L</code>	$\min L$	<code>\inf s</code>	$\inf s$
<code>\sup t</code>	$\sup t$	<code>\exp\!t</code>	$\exp t$	<code>\ln X</code>	$\ln X$
<code>\lg X</code>	$\lg X$	<code>\log X</code>	$\log X$	<code>\log_\alpha X</code>	$\log_\alpha X$
<code>\ker x</code>	$\ker x$	<code>\deg x</code>	$\deg x$	<code>\gcd(T,U,V,W,X)</code>	$\gcd(T,U,V,W,X)$
<code>\Pr x</code>	$\Pr x$	<code>\det x</code>	$\det x$	<code>\hom x</code>	$\hom x$
<code>\arg x</code>	$\arg x$	<code>\dim x</code>	$\dim x$	<code>\lim_{t\rightarrow n}T</code>	$\lim_{t\rightarrow n}T$

同余

语法	效果	语法	效果
<code>\pmod{m}</code>	$(\operatorname{mod} m)$	<code>a \bmod b</code>	$a \operatorname{mod} b$

微分

语法	效果	语法	效果	语法	效果
<code>\nabla</code>	∇	<code>\partial x</code>	∂x	<code>\mathrm{d}x</code>	$\mathrm{d}x$
<code>\dot x</code>	$\dot x$	<code>\ddot y</code>	$\ddot y$		

集合

语法	效果	语法	效果	语法	效果	语法	效果	语法	效果
<code>\forall</code>	\forall	<code>\exists</code>	\exists	<code>\empty</code>	\empty	<code>\emptyset</code>	\emptyset	<code>\varnothing</code>	\varnothing
<code>\in</code>	\in	<code>\ni</code>	\ni	<code>\notin</code>	\notin	<code>\notin</code>	\notin	<code>\subset</code>	\subset
<code>\subseteq</code>	\subseteq	<code>\supset</code>	\supset	<code>\supseteq</code>	\supseteq	<code>\cap</code>	\cap	<code>\bigcap</code>	\bigcap
<code>\cup</code>	\cup	<code>\bigcup</code>	\bigcup	<code>\biguplus</code>	\biguplus	<code>\sqsubset</code>	\sqsubset	<code>\sqsubseteq</code>	\sqsubseteq
<code>\sqsupset</code>	\sqsupset	<code>\sqsupseteq</code>	\sqsupseteq	<code>\sqcap</code>	\sqcap	<code>\sqcup</code>	\sqcup	<code>\bigsqcup</code>	\bigsqcup

逻辑

语法	效果	语法	效果	语法	效果	语法	效果
p	<i>p</i>	\land	\wedge	\wedge	\wedge	\bigwedge	\bigwedge
\bar{q}	\bar{q}	\lor	\vee	\vee	\vee	\bigvee	\bigvee
\to p	$\rightarrow p$						
\lnot	\neg	\neg q	$\neg q$	\setminus	\setminus	\smallsetminus	\smallsetminus

根号

语法	效果	语法	效果
\sqrt{3}	$\sqrt{3}$	\sqrt[n]{3}	$\sqrt[n]{3}$

关系符号

语法	效果
\Delta ABC\sim\Delta XYZ	$\Delta ABC \sim \Delta XYZ$
\sqrt{3}\approx 1.732050808\ldots	$\sqrt{3} \approx 1.732050808 \dots$

<code>\simeq</code>	\simeq
<code>\cong</code>	\cong
<code>\dot{=}</code>	$\dot{=}$
<code>\ggg</code> \ggg	\ggg
<code>\gg</code> \gg	\gg
	$>$
<code>\ge</code>	\geq
<code>\geqq</code>	\geqq
<code>=</code>	$=$
<code>\leq</code>	\leq
<code>\leqq</code>	\leqq
<code><</code>	$<$
<code>\ll</code>	\ll

<code>\lll</code>	\lll
<code>(x-y)^2\equiv(-x+y)^2\equiv x^2-2xy+y^2</code>	$(x-y)^2 \equiv (-x+y)^2 \equiv x^2 - 2xy + y^2$
<code>\begin{align}</code> <code>\because\begin{cases}</code> <code>\acute{a}x^2+bx^2+c\gtrless0\gtrless\grave{a}x^2+bx^2+c\\</code> <code>\acute{a}>0>\grave{a}</code> <code>\end{cases}\\</code> <code>\therefore\frac{-b\pm\sqrt{b^2-4\acute{a}c}}{2\acute{a}}</code> <code>{_lessgtr^{\gtrlessx_lessgtr^{\gtrless\frac{-b\pm\sqrt{b^2-4\grave{a}c}}{2\grave{a}}}}}</code> <code>\end{align}</code>	$\therefore \begin{cases} \acute{a}x^2 + bx^2 + c \gtrless 0 \gtrless \grave{a}x^2 + bx^2 + c \\ \acute{a} > 0 > \grave{a} \end{cases}$ $\therefore \frac{-b \pm \sqrt{b^2 - 4\acute{a}c}}{2\acute{a}} \gtrless \frac{-b \pm \sqrt{b^2 - 4\grave{a}c}}{2\grave{a}}$
<code>x\not\equiv N</code>	$x \not\equiv N$
<code>x\ne A</code>	$x \neq A$
<code>x\neq C</code>	$x \neq C$

t\propto v	$t \propto v$
\pm	±
\mp	∓

几何符号

特征		语法	效果
菱形		\Diamond	◇
正方形		\Box	□
三角形	Delta	\Delta	Δ
	图型	\triangle	△
角名		\angle\Alpha\Beta\Gamma	∠ABΓ
角度		\sin\!\frac{\pi}{3}=\sin60^\circ=\frac{\sqrt{3}}{2}	$\sin\frac{\pi}{3} = \sin 60^\circ = \frac{\sqrt{3}}{2}$
垂直		\perp	⊥

箭头符号

语法	效果	语法	效果	语法	效果
\leftarrow	←	\gets	←	\rightarrow	→
\to	→	\leftrightarrow	↔	\longleftarrow	←
\longrightarrow	→	\mapsto	↦	\longmapsto	↦
\hookrightarrow	↪	\hookleftarrow	↩	\nearrow	↗
\searrow	↘	\swarrow	↙	\nwarrow	↖
\uparrow	↑	\downarrow	↓	\updownarrow	↕

语法	效果	语法	效果	语法	效果	语法
\rightharpoonup	↷	\rightharpoondown	↘	\leftharpoonup	↶	\leftharpoondown
\upharpoonleft	↰	\upharpoonright	↱	\downharpoonleft	↴	\downharpoonright

语法	效果	语法	效果	语法	效果
----	----	----	----	----	----

<code>\Leftarrow</code>	\Leftarrow	<code>\Rightarrow</code>	\Rightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\Longleftarrow</code>	\Longleftarrow	<code>\Longrightarrow</code>	\Longrightarrow	<code>\Longleftrightarrow</code> (or <code>\iff</code>)	\Longleftrightarrow
<code>\Uparrow</code>	\Uparrow	<code>\Downarrow</code>	\Downarrow	<code>\Updownarrow</code>	\Updownarrow

特殊符号

语法	效果	语法	效果	语法	效果	语法	效果	语法	效果	语法	效果
<code>\eth</code>	\eth	<code>\S</code>	\S	<code>\P</code>	\P	<code>\%</code>	$\%$	<code>\dagger</code>	\dagger	<code>\ddagger</code>	\ddagger
<code>\star</code>	\star	<code>*</code>	$*$	<code>\ldots</code>	\ldots	<code>\smile</code>	\smile	<code>\frown</code>	\frown	<code>\wr</code>	\wr

语法	效果	语法	效果	语法	效果
<code>\oplus</code>	\oplus	<code>\bigoplus</code>	\bigoplus	<code>\otimes</code>	\otimes
<code>\bigotimes</code>	\bigotimes	<code>\times</code>	\times	<code>\cdot</code>	\cdot
<code>\div</code>	\div	<code>\circ</code>	\circ	<code>\bullet</code>	\bullet
<code>\bigodot</code>	\bigodot	<code>\boxtimes</code>	\boxtimes	<code>\boxplus</code>	\boxplus

语法	效果	语法	效果	语法	效果	语法	效果
<code>\triangleleft</code>	\triangleleft	<code>\triangleright</code>	\triangleright	<code>\infty</code>	∞	<code>\bot</code>	\bot
<code>\top</code>	\top	<code>\vdash</code>	\vdash	<code>\vDash</code>	\vDash	<code>\Vdash</code>	\Vdash
<code>\models</code>	\models	<code>\lVert</code>	\lVert	<code>\rVert</code>	\rVert		

语法	效果	语法	效果	语法	效果
<code>\imath</code>	\imath	<code>\hbar</code>	\hbar	<code>\ell</code>	ℓ
<code>\mho</code>	\mho	<code>\Finv</code>	\Finv	<code>\Re</code>	\Re
<code>\Im</code>	\Im	<code>\wp</code>	\wp	<code>\complement</code>	\complement

语法	效果	语法	效果	语法	效果	语法	效果
<code>\diamondsuit</code>	\diamondsuit	<code>\heartsuit</code>	\heartsuit	<code>\clubsuit</code>	\clubsuit	<code>\spadesuit</code>	\spadesuit
<code>\Game</code>	\Game	<code>\flat</code>	\flat	<code>\natural</code>	\natural	<code>\sharp</code>	\sharp

上标、下标及积分等

功能	语法	效果
上标	<code>a^2</code>	a^2
下标	<code>a_2</code>	a_2
组合	<code>a^{2+2}</code>	a^{2+2}
	<code>a_{i,j}</code>	$a_{i,j}$
结合上下标	<code>x_2^3</code>	x_2^3
前置上下标	<code>{_1^2\!x_3^4}</code>	${}_1^2X_3^4$
导数 (HTML)	<code>x'</code>	x'
导数 (PNG)	<code>x^\prime</code>	x'
导数 (错误)	<code>x\prime</code>	$x/$
导数点	<code>\dot{x}</code>	\dot{x}
	<code>\ddot{y}</code>	\ddot{y}

向量	<code>\vec{c}</code>	\vec{c}
	<code>\overleftarrow{a b}</code>	\overleftarrow{ab}
	<code>\overrightarrow{c d}</code>	\overrightarrow{cd}
	<code>\widehat{e f g}</code>	\widehat{efg}
上弧 (注: 正确应该用 <code>\overarc</code> , 但在这里行不通。 要用建议的语法作为解决办法)	<code>\overset{\frown}{AB}</code>	$\overset{\frown}{AB}$
上划线	<code>\overline{h i j}</code>	\overline{hij}
下划线	<code>\underline{k l m}</code>	\underline{klm}
上括号	<code>\overbrace{1+2+\cdots+100}</code>	$\overbrace{1+2+\cdots+100}$
	<code>\begin{matrix} 5050 \\ \overbrace{1+2+\cdots+100} \\ \end{matrix}</code>	$\begin{matrix} 5050 \\ \overbrace{1+2+\cdots+100} \\ \end{matrix}$
下括号	<code>\underbrace{a+b+\cdots+z}</code>	$\underbrace{a+b+\cdots+z}$

	$\begin{matrix} \underbrace{a+b+\cdots+z}_{26} \\ \\ \end{matrix}$	
求和	$\sum_{k=1}^N k^2$	$\sum_{k=1}^N k^2$
	$\begin{matrix} \sum_{k=1}^N k^2 \\ \end{matrix}$	$\sum_{k=1}^N k^2$
求积	$\prod_{i=1}^N x_i$	$\prod_{i=1}^N x_i$
	$\begin{matrix} \prod_{i=1}^N x_i \\ \end{matrix}$	$\prod_{i=1}^N x_i$
上积	$\coprod_{i=1}^N x_i$	$\coprod_{i=1}^N x_i$
	$\begin{matrix} \coprod_{i=1}^N x_i \\ \end{matrix}$	$\coprod_{i=1}^N x_i$
极限	$\lim_{n \rightarrow \infty} x_n$	$\lim_{n \rightarrow \infty} x_n$
	$\lim_{n \rightarrow \infty} x_n$	$\lim_{n \rightarrow \infty} x_n$

	<code>\infty x_n \end{matrix}</code>	
积分	<code>\int_{-N}^N e^x \, dx</code>	$\int_{-N}^N e^x dx$
	<code>\begin{matrix} \int_{-N}^N e^x \, dx \end{matrix}</code>	$\int_{-N}^N e^x dx$
双重积分	<code>\iint_D \, dx \, dy</code>	$\iint_D dx dy$
三重积分	<code>\iiint_E \, dx \, dy \, dz</code>	$\iiint_E dx dy dz$
四重积分	<code>\iiiiint_F \, dx \, dy \, dz \, dt</code>	$\iiint_F dx dy dz dt$
闭合的 曲线 、 曲面积分	<code>\oint_C x^3 \, dx + 4y^2 \, dy</code>	$\oint_C x^3 dx + 4y^2 dy$
交集	<code>\bigcap_{1}^n p</code>	$\bigcap_1^n p$
并集	<code>\bigcup_{1}^k p</code>	$\bigcup_1^k p$

分数、 矩阵 和多行列式

功能	语法	效果
----	----	----

分数	<code>\frac{2}{4}=0.5</code>	$\frac{2}{4} = 0.5$
小型分数	<code>\tfrac{2}{4} = 0.5</code>	$\tfrac{2}{4} = 0.5$
大型分数 (嵌套)	<code>\cfrac{2}{c + \cfrac{2}{d + \cfrac{2}{4}}} = a</code>	$\frac{2}{c + \frac{2}{d + \frac{2}{4}}} = a$
大型分数 (不嵌套)	<code>\dfrac{2}{4} = 0.5 \quad \dfrac{2}{c + \dfrac{2}{d + \dfrac{2}{4}}} = a</code>	$\frac{2}{4} = 0.5 \quad \frac{2}{c + \frac{2}{d + \frac{2}{4}}} = a$
二项式 系数	<code>\dbinom{n}{r}=\binom{n}{n-r}</code> <code>r}=C^n_r=C^n_{n-r}</code>	$\binom{n}{r} = \binom{n}{n-r} = C^n_r = C^n_{n-r}$
小型 二项式 系数	<code>\tbinom{n}{r}=\tbinom{n}{n-r}</code> <code>r}=C^n_r=C^n_{n-r}</code>	$\binom{n}{r} = \binom{n}{n-r} = C^n_r = C^n_{n-r}$
大型 二项式 系数	<code>\binom{n}{r}=\dbinom{n}{n-r}</code> <code>r}=C^n_r=C^n_{n-r}</code>	$\binom{n}{r} = \binom{n}{n-r} = C^n_r = C^n_{n-r}$
矩阵	<code>\begin{matrix}</code> <code>x & y \\</code> <code>z & v</code> <code>\end{matrix}</code>	$\begin{matrix} x & y \\ z & v \end{matrix}$

<pre>\begin{vmatrix} x & y \\ z & v \end{vmatrix}</pre>	$\begin{vmatrix} x & y \\ z & v \end{vmatrix}$
<pre>\begin{Vmatrix} x & y \\ z & v \end{Vmatrix}</pre>	$\begin{Vmatrix} x & y \\ z & v \end{Vmatrix}$
<pre>\begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}</pre>	$\begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}$
<pre>\begin{Bmatrix} x & y \\ z & v \end{Bmatrix}</pre>	$\begin{Bmatrix} x & y \\ z & v \end{Bmatrix}$
<pre>\begin{pmatrix} x & y \\ z & v \end{pmatrix}</pre>	$\begin{pmatrix} x & y \\ z & v \end{pmatrix}$
<pre>\bigl(\begin{smallmatrix} a&b\\ c&d \end{smallmatrix} \bigr)</pre>	$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

条件定义	<pre>f(n) = \begin{cases} n/2, & \text{\mbox{if }n\mbox{ is even} } \\ 3n+1, & \text{\mbox{if }n\mbox{ is odd} } \end{cases}</pre>	$f(n) = \begin{cases} n/2, & \text{if } n \text{ is even} \\ 3n+1, & \text{if } n \text{ is odd} \end{cases}$
多行等式	<pre>\begin{align} f(x) &= (m+n)^2 \\ &= m^2+2mn+n^2 \end{align}</pre>	$\begin{aligned} f(x) &= (m+n)^2 \\ &= m^2+2mn+n^2 \end{aligned}$
	<pre>\begin{alignat}{2} f(x) &= (m-n)^2 \\ f(x) &= (-m+n)^2 \\ &= m^2-2mn+n^2 \end{alignat}</pre>	$\begin{aligned} f(x) &= (m-n)^2 \\ f(x) &= (-m+n)^2 \\ &= m^2-2mn+n^2 \end{aligned}$
多行等式 (左对齐)	<pre>\begin{array}{lcl} z & = & a \\ f(x,y,z) & = & x + y + z \end{array}</pre>	$\begin{array}{lcl} z & = & a \\ f(x,y,z) & = & x + y + z \end{array}$
多行等式 (右对齐)	<pre>\begin{array}{lcr} z & = & a \\ f(x,y,z) & = & x + y + z \end{array}</pre>	$\begin{array}{lcr} z & = & a \\ f(x,y,z) & = & x + y + z \end{array}$
长公式换行	<pre><math>f(x) \ ,\,!\</math> <math>= \sum_{n=0}^{\infty} a_n x^n \</math> <math>= a_0+a_1x+a_2x^2+\cdots\</math></pre>	$\begin{aligned} &f(x) \\ &= \sum_{n=0}^{\infty} a_n x^n \end{aligned}$

		$= a_0 + a_1x + a_2x^2 + \cdots$															
方程组	<pre>\begin{cases}3x + 5y + z \\7x - 2y + 4z \\-6x + 3y + 2z\end{cases}</pre>	$\begin{cases}3x + 5y + z \\7x - 2y + 4z \\-6x + 3y + 2z\end{cases}$															
数组	<pre>\begin{array}{ c c c } \hline a & b & S \\ \hline 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ \hline \end{array}</pre>	<table><tr><td><i>a</i></td><td><i>b</i></td><td><i>S</i></td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	<i>a</i>	<i>b</i>	<i>S</i>	0	0	1	0	1	1	1	0	1	1	1	0
<i>a</i>	<i>b</i>	<i>S</i>															
0	0	1															
0	1	1															
1	0	1															
1	1	0															

字体

希腊字母

斜体小写希腊字母一般用于在方程中显示变量。

正体希腊字母

特征	语法	效果	注释 / 外部链接
大写 字母	<code>\Alpha \Beta \Gamma \Delta</code> <code>\Epsilon \Zeta \Eta \Theta</code>	ΑΒΓΔΕΖΗΘ	Α Β Γ Δ Ε Ζ Η Θ
	<code>\Iota \Kappa \Lambda \Mu \Nu \Xi</code> <code>\Omicron \Pi</code>	ΙΚΑΜΝΞΟΠ	Ι Κ Λ Μ Ν Ξ Ο Π
	<code>\Rho \Sigma \Tau \Upsilon \Phi</code> <code>\Chi \Psi \Omega</code>	ΡΣΤΥΦΧΨΩ	Ρ Σ Τ Υ Φ Χ Ψ Ω
小写 字母	<code>\alpha \beta \gamma \delta</code> <code>\epsilon \zeta \eta \theta</code>	$\alpha\beta\gamma\delta\epsilon\zeta\eta\theta$	
	<code>\iota \kappa \lambda \mu \nu</code> <code>\xi \omicron \pi</code>	$\iota\kappa\lambda\mu\nu\xi\omicron\pi$	
	<code>\rho \sigma \tau \upsilon \phi</code> <code>\chi \psi \omega</code>	$\rho\sigma\tau\upsilon\phi\chi\psi\omega$	
异体 字母	<code>\Epsilon \epsilon \varepsilon</code>	Εεε	
	<code>\Theta \theta \vartheta</code>	Θθθ	

	<code>\Kappa\kappa\varkappa</code>	$\text{K}\kappa\text{K}$	
	<code>\Pi\pi\varpi</code>	$\text{P}\pi\varpi$	
	<code>\Rho\rho\varrho</code>	$\text{P}\rho\varrho$	
	<code>\Sigma\sigma\varsigma</code>	$\Sigma\sigma\varsigma$	
	<code>\Phi\phi\varphi</code>	$\Phi\phi\varphi$	
已停用字母	<code>\digamma</code>	F	F [1]

粗体希腊字母		
特征	语法	效果
大写字母	<code>\boldsymbol{\Alpha \Beta \Gamma \Delta \Epsilon \Zeta \Eta \Theta}</code>	ΑΒΓΔΕΖΗΘ
	<code>\boldsymbol{\Iota \Kappa \Lambda \Mu \Nu \Xi \Omicron \Pi}</code>	ΙΚΑΜΝΞΟΠ

	<code>\boldsymbol{\Rho \Sigma \Tau \Upsilon \Phi}</code> <code>\Chi \Psi \Omega</code>	$\rho \sigma \tau \upsilon \phi \chi \psi \omega$
小写字母	<code>\boldsymbol{\alpha \beta \gamma \delta \epsilon \zeta \eta \theta}</code> <code>\epsilon \zeta \eta \theta</code>	$\alpha \beta \gamma \delta \epsilon \zeta \eta \theta$
	<code>\boldsymbol{\iota \kappa \lambda \mu \nu \xi \omicron \pi}</code>	$\iota \kappa \lambda \mu \nu \xi \omicron \pi$
	<code>\boldsymbol{\rho \sigma \tau \upsilon \phi \chi \psi \omega}</code> <code>\chi \psi \omega</code>	$\rho \sigma \tau \upsilon \phi \chi \psi \omega$
异体字母	<code>\boldsymbol{\Epsilon \epsilon \varepsilon}</code>	$\mathbf{E} \epsilon \epsilon$
	<code>\boldsymbol{\Theta \theta \vartheta}</code>	$\mathbf{\Theta} \theta \vartheta$
	<code>\boldsymbol{\Kappa \kappa \varkappa}</code>	$\mathbf{K} \kappa \kappa$
	<code>\boldsymbol{\Pi \pi \varpi}</code>	$\mathbf{\Pi} \pi \varpi$
	<code>\boldsymbol{\Rho \rho \varrho}</code>	$\mathbf{P} \rho \varrho$
	<code>\boldsymbol{\Sigma \sigma \varsigma}</code>	$\mathbf{\Sigma} \sigma \varsigma$
	<code>\boldsymbol{\Phi \phi \varphi}</code>	$\mathbf{\Phi} \phi \varphi$

已停用字母	<code>\boldsymbol{\digamma}</code>	
-------	------------------------------------	--

黑板粗体

语法

```
\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}
```

效果



黑板粗体（ `Blackboardbold` ）一般用于表示数学和物理学中的向量或集合的符号。 备注：

- 1. `{花括号}`中只有使用大写拉丁字母才能正常显示，使用小写字母或数字会得到其他符号。

正粗体

语法

```
\mathbf{012...abc...ABC...}
```

效果

0 1 2 3 4 5 6 7 8 9



A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

备注

花括号 {} 内只能使用拉丁字母和数字，不能使用希腊字母如 \alpha 等。斜粗体

语法

```
\boldsymbol{012...abc...ABC...\alpha \beta\gamma...}
```

效果

0 1 2 3 4 5 6 7 8 9

a b c d e f g h i j k l m n o p q r s t u v w x y z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

$\alpha \beta \gamma \delta \epsilon \zeta \eta \theta \iota \kappa \lambda \mu \nu \xi \omicron \pi \rho \sigma \tau \upsilon \phi \chi \psi \omega$

备注

使用 `\boldsymbol{}` 可以加粗所有合法的符号。

斜体数字

语法

`\mathit{0123456789}`

效果

0123456789

罗马体

语法

`\mathrm{012...abc...ABC...}`或`\mbox{}`或`\operatorname{}`

效果

0123456789

ABCDEFGHIJKLMNOPQRSTUVWXYZ

abcdefghijklmnopqrstuvwxyz

备注

罗马体可以使用数字和 [拉丁字母](#) 。

哥特体

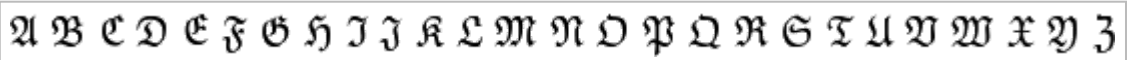
语法

```
\mathfrak{012...abc...ABC...}
```

效果

0 1 2 3 4 5 6 7 8 9

a b c d e f g h i j k l m n o p q r s t u v w x y z



备注

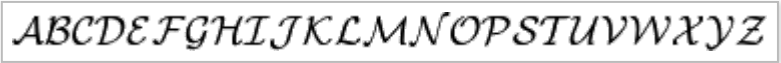
哥特体可以使用数字和拉丁字母。

手写体

语法

```
\mathcal{ABC...}
```

效果



备注

手写体仅对大写拉丁字母有效。

希伯来字母

语法

```
\aleph\beth\gimel\daleth
```

效果

效果

括号

功能	语法	显示
不好看	<code>(\frac{1}{2})</code>	$(\frac{1}{2})$
好看了	<code>\left(\frac{1}{2}\right)</code>	$\left(\frac{1}{2}\right)$

您可以使用 `\left` 和 `\right` 来显示不同的括号：

功能	语法	显示
圆括号，小括号	<code>\left(\frac{a}{b}\right)</code>	$\left(\frac{a}{b}\right)$
方括号，中括号	<code>\left[\frac{a}{b}\right]</code>	$\left[\frac{a}{b}\right]$

花括号，大括号	<code>\left\{\frac{a}{b}\right\}</code>	$\left\{\frac{a}{b}\right\}$
角括号	<code>\left\langle\frac{a}{b}\right\rangle</code>	$\left\langle\frac{a}{b}\right\rangle$
单竖线，绝对值	<code>\left \frac{a}{b}\right </code>	$\left \frac{a}{b}\right $
双竖线，范	<code>\left\ \frac{a}{b}\right\ </code>	$\left\ \frac{a}{b}\right\ $
取整函数 (Floor function)	<code>\left\lfloor\frac{a}{b}\right\rfloor</code>	$\left\lfloor\frac{a}{b}\right\rfloor$
取顶函数 (Ceiling function)	<code>\left\lceil\frac{c}{d}\right\rceil</code>	$\left\lceil\frac{c}{d}\right\rceil$
斜线与反斜线	<code>\left/\frac{a}{b}\backslash</code>	$\left/\frac{a}{b}\backslash$
上下箭头	<code>\left\uparrow\frac{a}{b}\downarrow</code>	$\left\uparrow\frac{a}{b}\downarrow$

	<code>\left \Uparrow \frac{a}{b} \right \Downarrow</code>	$\Uparrow \frac{a}{b} \Downarrow$
	<code>\left \updownarrow \frac{a}{b} \right \Updownarrow</code>	$\updownarrow \frac{a}{b} \Updownarrow$
混合括号	<code>\left [0,1 \right)</code> <code>\left \langle \psi \right </code>	$[0, 1)$ $\langle \psi $
单左括号	<code>\left \{\frac{a}{b} \right .</code>	$\left\{ \frac{a}{b} \right.$
单右括号	<code>\left . \frac{a}{b} \right \}</code>	$\left. \frac{a}{b} \right\}$

备注：

- 可以使用 `\big, \Big, \bigg, \Bigg` 控制括号的大小，比如代码

```
\Bigg ( \bigg [ \Big \{\big\langle \left | \right | \frac{a}{b} \right | \right)
| \big \rangle \Big\} \bigg ] \Bigg )
```

显示：

$$\left(\left[\left\{\left\langle \right| \right| x \right| \right\rangle\right]\right)$$

空格

注意 TEX 能够自动处理大多数的空格，但是您有时候需要自己来控制。

功能	语法	显示	宽度
2 个 quad 空格	<code>\alpha\qquad\beta</code>	$\alpha \quad \beta$	$2m$
quad 空格	<code>\alpha\quad\beta</code>	$\alpha \quad \beta$	m
大空格	<code>\alpha\, \beta</code>	$\alpha \beta$	$\frac{m}{3}$
中等空格	<code>\alpha\;\beta</code>	$\alpha \beta$	$\frac{2m}{7}$
小空格	<code>\alpha\,\beta</code>	$\alpha \beta$	$\frac{m}{6}$
没有空格	<code>\alpha\beta</code>	$\alpha\beta$	0
紧贴	<code>\alpha\!\!\beta</code>	$\alpha\beta$	$-\frac{m}{6}$

颜色

语法

- 字体颜色： `{\color{色调}表达式}`
- 背景颜色： `{\pagecolor{色调}表达式}`

支援色调表

Colors supported

Apricot	Aquamarine	Bittersweet	Black
Blue	BlueGreen	BlueViolet	BrickRed
Brown	BurntOrange	CadetBlue	CarnationPink
Cerulean	CornflowerBlue	Cyan	Dandelion
DarkOrchid	Emerald	ForestGreen	Fuchsia
Goldenrod	Gray	Green	GreenYellow
JungleGreen	Lavender	LimeGreen	Magenta
Mahogany	Maroon	Melon	MidnightBlue
Mulberry	NavyBlue	OliveGreen	Orange

OrangeRed	Orchid	Peach	Periwinkle
PineGreen	Plum	ProcessBlue	Purple
RawSienna	Red	RedOrange	RedViolet
Rhodamine	RoyalBlue	RoyalPurple	RubineRed
Salmon	SeaGreen	Sepia	SkyBlue
SpringGreen	Tan	TealBlue	Thistle
Turquoise	Violet	VioletRed	White
WildStrawberry	Yellow	YellowGreen	YellowOrange

★注：输入时第一个字母必需以大写输入，如 `\color{OliveGreen}` 。

例子

- $$\{\color{Blue}x^2\} + \{\color{Brown}2x\} - \{\color{OliveGreen}1\}$$

$$\color{Blue}x^2 + \color{Brown}2x - \color{Green}1$$

- $$x_{\{\color{Maroon}1,2\}} = \frac{-b \pm \sqrt{\{\{\color{Maroon}b^2 - 4ac\}\}}}{2a}$$

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$


小型数学公式

当要把分数等公式放进文字中的时候，我们需要使用小型的数学公式。

苹果原产于欧洲和中亚细亚。哈萨克的阿拉木图与新疆阿力麻里有苹果城的美誉。中国古代的林檎、柰、花红等水果被认为是中国土生苹果品种或与苹果相似的水果。苹果在中国

的栽培记录可以追溯至西汉时期，汉武帝时，10 的 $f(x) = 5 + \frac{1}{5}$ 是 2。上林苑中曾栽

培林檎和柰，当时多用于薰香衣裳等，亦有置于床头当香熏或置于衣服初作为香囊，总之一般不食用。但也有看法认为，林檎和柰是现在的沙果，曾被误认为苹果，真正意义上的苹果是元朝时期从中亚地区传入中国，当时只有在宫廷才可享用。

-  并不好看。

苹果原产于欧洲和中亚细亚。哈萨克的阿拉木图与新疆阿力麻里有苹果城的美誉。中国古代的林檎、柰、花红等水果被认为是中国土生苹果品种或与苹果相似的水果。苹果在中国的栽培记录可以追溯至西汉时期，汉武帝时，10 的 $f(x)=5+\frac{1}{5}$ 是 2。上林苑中曾栽培林檎

和柰，当时多用于薰香衣裳等，亦有置于床头当香熏或置于衣服初作为香囊，总之一般不食用。但也有看法认为，林檎和柰是现在的沙果，曾被误认为苹果，真正意义上的苹果是元朝时期从中亚地区传入中国，当时只有在宫廷才可享用。

- ✓ 好看些了。

可以使用

```
\begin{smallmatrix}...\end{smallmatrix}
```

或直接使用 `{{ Smallmath }}` 模板。

```
{{Smallmath|f= f(x)=5+\frac{1}{5} }}
```

强制使用 PNG

假设我们现在需要一个 PNG 图的数学公式。

若输入 $2x=1$ 的话：

$$2x = 1$$

↑ 这并不是我们想要的。

若你需要强制输出一个 PNG 图的数学公式的话，你可于公式的最后加上 `\,`（小空格，但于公式的最后是不会显示出来）。

输入 $2x=1 \,$ 的话：

$$2x = 1$$

↑ 以 PNG 图输出。

你也可以使用 `\,!\`，这个亦能强制使用 PNG 图像。

阅读更多：[Help:Displaying a formula#Forced PNG rendering](#)

|TeX 各版本概述及基本约定，特殊字符|

+-----+

tex 提供 300 多条基本排版命令

由 D.E.Knuth1978 年开发

plain tex：在 tex 基础上新定义 600 多条复合命令

AMS-TEX：美国数学会开发（amsmath 宏包）排版的数学公式

LATEX：L.Lamport（1985）编写，适合排版普通文章和书籍

LATEX2e：可加载 amsmath 宏包，目前最流行的 TEX 宏包

版本：LATEX2.09-->LATEX2e-->LATEX3（开发中）

中文排版：

CCT：科学院张林波

TY（天元）：华师大肖刚、陈志杰教授开发

CJK：德国 W.Lemberg 开发，处理中日韩三国文字。

发行版 CTEX：集成了 CCT，TY，CJK 的 MikTEX 系统。

ChinaTEX: 内容涵盖 MiKTeX 系统及中文支持、常用外围软件、TeX\LaTeX 文档和模板
选萃等

TeX 中的长度

mm 毫米

cm 厘米

in 英寸 = 2.54cm = 72.27pt

pt 点

em 大写字母 M 的宽度

ex 小写字母 x 的高度

弹性长度：根据需要自动伸缩

正常值 plus 伸展值 minus 收缩值

实际长度可超过正常值和伸展值之和，但不能小于正常值和收缩值之差

`\documentclass[11pt]{article}` %11pt 字体，普通文章

% 导言区，全局命令

`\usepackage{CJK}` % 使用 CJK 宏包

`\begin{document}` % 主环境

`\begin{CJK}{GBK}{song}` % 汉字必须放入 CJK 环境

% 其它字体: song,kai,fs,hei,li,you

%CJK 的两种环境 CJK 和 CJK*

%GBK 是采用的字符集：GB，GBK，Bg5，Gbt

Hi,This is my first \LaTeX file

祝贺你，MikTeX 和 CJK 安装成功了

`\end{CJK}`

`\ent{document}`

CJK 和 CJK * 环境之间的切换

`\CJKspace` 和 `\CJKnospace`

`\CJKtilde` 重新定义~的长度

基本约定

分组 `{.....}`

注释符: `%`

西文标点后要加空格

各种环境的开始和结束命令最好独占一行

换行: 连续两个回车, 一个仅为空格

输入特殊字符

前加 `\` 的有: `#$%{}~_^&`

`\=\textbackslash`

`| , < , > = $ | $, $ < $, $ > $, $ * $` (中间星)

`* = *` 上面星

特殊符号 `\ s \ p \ dag \ ddag \ copyrigh` 版权号, `\ pounds` 磅

`\TeX \LaTeX \LaTeXe \AmS-\LaTeX` (最后这个要 `amsmath` 宏包)

单引号: ``` (1 键前面的)

双引号: ````

+-----+

| 段落、行距、水平间距、缩进命令 |

+-----+

强制分行：\\ 或 *[和下行间距离]、\newline

建议分行：\linebreak[0-4，数越大建议力度越大]

\linebreak 增加字间距，强制换行

\nolinebreak [n] 建议不分行

\mbox {内容} 内容保持在同一行

分段：\par，或两个回车

分页：自动分页

强制分页：\newpage

建议分页：\pagebreak [n] ， \nopagebreak [n]

增加当前页高度有时可以避免难看的分页

\enlargethispage {尺寸} 可增加的最大高度

\enlargethispage* {尺寸} 指定增加高度

水平间距：

\quad 插入空白相当于当前字体大小

\qqquad = \quad \times 2

\, , = \quad \times 3/18

\sim = ??? 好象比 \ 小

`\hspace{宽度大小}`, `\hspace*{宽度大小}`

`\hfill` 弹性长度: `\hspace{\hfill}` 插入空白, 撑满整行

`\hphantom{文本内容}`, 占据文本内容的宽度

`\vphantom[文本内容]`, ``

导引线: `\dotfill`, `\hrulefill`

垂直间距:

`\vspace{高度}` 和 `\vspace*{高度}`

`\vfill`: 相当于 `\vspace{\fill}`

`\smallskip:-->\vspace{\smallskipamount}`

`\medskip:-->\vspace{\medskipamount}`

`\bigskip:-->\vspace{\bigskipamount}`

段落首行缩进:

`\setlength{\parindent}{2em}`, 2 个 M

`\setlength{\parindent}{0pt}`, 首行不缩进

`\CJKindent`: 两个汉字

`\indent` 与 `\noindent`

每节的第一段首行不会自动缩进

`\hspace{\parindent}`

在导言区加入: `\usepackage{indentfirst}`

段落间距: `\lineskip+\parskip`

可用 `\setlength` 修改以上值

行距：

`\baselineskip`：相邻两行基线间距离

`\baselinestretch`：伸展因子

修改改变行距：`\renewcommand{\baselinestretch}[1.2]`

放在 `\begin{document}` 之后，字体尺寸改变时才生效

+-----+

| 西文字体属性、及中文字体属性、命令 |

+-----+

字体有 5 种属性

族：

`\rmfamily`：罗马字体

`\sffamily`：无衬线字体

`\ttfamily`：打字机字体

形状：（shape）直立斜

`\upshape`：直立

`\itshape`：意大利斜体

`\slshape`：斜体

`\scshape`：小体大写

系列：（series）宽度黑度

`\mdseries`：中等权重（黑）

`\bfseries`: 粗体

`\normalfont`: 默认字体, 中等权重直立罗马字体

西文尺寸: 10pt, 11pt, 12pt

`\em` 强调, 一般为斜体

以上命令称为声明 (无参数)

每一声明对应字体命令

命令只对其参数中的文本起作用

族:

`\testrm{}`, `\testsf{}`, `\testtt{}`

形状

`\textup{}`, `\textit{}`, `\textsl{}`, `\textsc{}`

系列

`\textmd{}`, `\textbf{}`

默认值: `\textnormal{}`

强调: `\emph{}`

定义了基本尺寸后, 可使用下面的声明来改变字体

`\tiny` 5pt

`\scriptsize` 7pt

`\footnotesize` 8pt

`\small` 9pt

`\normalsize` 10pt

`\large` 12pt

`\Large` 14.4pt

`\LARGE` 17.28pt

`\huge` 20.74pt

`\Huge` 24.88pt

以上尺寸是基本尺寸为 10pt 时的大小

所有西文字体命令对中文同样起作用

`italic` 和 `slanted` 斜体对中文一样

中文书籍基本字号为 5 号约等于 11pt

使用 GBK 编码时，可同时输入简繁体。

自定义字体大小

`\fontsize{字体尺寸}{行距}` 后面须加上 `\selectfont` 才生效

可得到任意大小汉字，西文最大不能超 `\Huge`

行距通常为字体大小的 1.2–1.5 倍

例： `\fontsize{12pt}{\baselineskip}\selectfont`

`\usepackage{type1cm}` 任意大小西文

行距： `\linespread{1.3}` 产生 1.5 倍行距，1.6 产生双倍行距，效果夸张，不适合出版

用下面方法： `\setlength{\baselineskip}{1.5\baselineskip}{.....}`

正文中更换字体：\CJKfamily{字体族}

为方便，作以下自定义

```
\newcommand*{\SONG}{\CJKfamily{song}}
```

```
\newcommand*{\HEI}{\CJKfamily{hei}}
```

```
\newcommand*{\KAI}{\CJKfamily{kai}}
```

```
\newcommand*{\FS}{\CJKfamily{fs}}
```

```
\newcommand*{\LI}{\CJKfamily{li}}
```

```
\newcommand*{\YOU}{\CJKfamily{YOU}}
```

引用更改为宋体：\SONG 正文

文本居中

文本默认为左对齐

单行文本居中命令

```
\centerline{....}
```

多行使用 \begin{center} 环境

居中声明：\centering（不建议使用）

```
+-----+
```

```
| 参考文献环境 |
```

```
+-----+
```

```
\begin{thebibliography}{编号样本}
```

`\bibitem[记号]{引用标志}` 文献条目

·
·
·

`\bibitem[记号]{引用标志}` 文献条目

`\end{thebibliography}`

`\bibitem[省略为方括号数字]{不可省由字母，数字和，号外符号组成，各个文献互不相同}` 文献条目

编号样本，指定记号的宽度，一般为数字

引用文献时用 `\ cite{引用标志 1, 引用标志 2}`

调节各文献间距离

`\setlength{\itemsep}{高度}`

标题缺省为左对齐 Reference

`(\Large\bfseries)`

由 `\ refname` 值确定

书籍参考文献标题由 `\ bibname` 值确定

文献条目包含：作者，标题，出版社，年代，版本，页码

一行放不下，后面自动缩进，距离为“编号样本”宽度

+-----+

| 脚 注 |

+-----+

\footnote{脚注内容}

article 文档，整篇同一编号

book 和 report 文档，每章统一编号

+-----+

| 文档类别、章节 |

+-----+

\documentclass[可选项]{article,book,report,letter, 只能选一种}

article 短篇文章

report 长篇报告，可分章

book （书）可含 chapter，奇偶页采用不同处理

可选项对整篇文档起作用，使用多个选项互不排斥，用逗号分隔

基本字体：10pt, 11pt, 12pt

纸张大小：letterpaper, a4paper, a5paper...

排版方向：portrait（竖向），landscape（横向）

标题页：titlepage, notitlepage

final 和 draft

指定纸张大小（只有 latex2e 中使用）

letterpaper (11*8.5in)

legalpaper (14*8.5in)

executivepaper (10.5*7.25in)

a4paper (29.7*21cm)

a5paper (21*14.8cm)

b5paper (25*17.6cm)

默认值是 letterpaper, 美国信纸尺寸, 纵向

landscape 横向

章节命令:

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

`\命令 [短标题]{标题}`, 用于显示在目录和页眉

`\命令 *{标题}`

层次号: book 中 `\part` 为 - 1 (article 中为 0), `\chapter` 为 0,

book 类

`\part, \chapter` 独立编号, 只出现在奇数页, `\part` 独占一页

`\section, \subsection` 关联编号

article 类

`\part, \section` 独立编号

`\subsection, \subsubsection` 关联编号

星号章节不参加自动编号

`\part` 和 `\chapter` 标题占两行

标题中文化

`\renewcommand{\partname}{}{}`

`\renewcommand{\thepart}{第 \, \Roman{part}}\, 篇}`

修改编号最深层次

```
\setcounter{secnumberdepth}{数}
```

secnumberdepth, book 类 - 2~5, 缺省为 2, -2 时取消编号

article 类 - 1~5, 缺省为 3

自动编号章节对应计数器

part, chapter, section, bsection,subsubsection,paragraph,subparagraph

修改计数器值:

```
\setcounter{计数器名}{数}
```

TeX 的版式首先是文件版式,也就是我们接触的的第一个最最最简单的 TeX 文档中的第一句:

```
\documentstyle[选项 1, 选项 2,...]{格式}
```

其中的 [] 及选项是可选的,而 {格式} 是必需的. LaTeX 有以下几种格式: article(文章), report(报告), book(书), letter(信件).

CCT 对前面三种进行了汉化,分别是 carticle, creport 和 cbook.

常用的选项有

10pt/11pt/12pt: 定义基准字号, 缺省 10pt.

twoside: 两面印刷格式, 因奇偶页装订位置不同而异, 缺省为单页.

twocolumn: 双栏排版, 缺省为单栏.

titlepage: 仅在 article 格式使用, 它是 \maketitle 产生一个单独的文章题目页, 同时 abstract 也产生单独页

leqno: 使公式标号出现在左边, 缺省在右

fleqn: 公式左对齐, 缺省中间对齐

thesis: 专用于毕业论文排版

IEEE: 专向 IEEE 类杂志投稿的格式

IFTHEN: 排版计算机程序

`\columnsep` 指定两栏间距

`\columnseprule` 两栏间竖线宽度, 默认为 0。

`\mathindent` 选择 fleqn 时 左边界的缩进量

以上参数修改用 `\setlength{\mathindent}{2.5cm}`

LaTeX 的一个页面有页眉 (head, 通常是杂志名, 卷号, 当前章节名等), 主体 (body, 正文, 包括脚注及图表), 页脚 (通常是页码, 如果页码放在页眉处, 则页脚可能是空的). 我们用下面的指令控制页版面:

`\pagestyle{参数}`

参数可以是

plain: 页码在页脚居中, 页眉空白, 为 article, report 的缺省

empty: 页眉页脚都空白

heading: 页脚为空, 页眉为正文的章节信息及页码

myheadings: 自定义页眉, 内容由

`\markboth{left_head}{right_head}` (双面 twoside 排版)

或 `\markright{right_head}` (单面排版)

`\pagestyle` 是对全文有效的, 而如果要定义某页的页面格式, 则可用 `\thispagestyle`, 用法同上, 下一页自动恢复原设置.

这里提到页码, 我们经常会修改页码的表现格式, 可能用阿拉伯数字或罗马数字等, 则修改 `\pagenumbering{参数}`, 这里参数可以是 `arabic`(阿拉伯数字), `roman`(小写罗马数字: i, ii, iii,) `Roman`(大写罗马数字: I, II, III,), `alph`(小写英文: a, b, c,), `Alph`(大写英文: A, B, C). 如果需要更改某页的页码, 则用修改计数器命令: `\setcounter{page}{数字}`, 这里 `page` 是页码计数器.

```
+-----+
|* 修改版面的命令 *|
+-----+
```

TeX 中还经常会在 `\documentstyle` 和 `\begin{document}` 之间常用到很多修改版面尺寸的命令. 如未指明, 这些命令的格式都是用 `\para_name=newvalue` 的格式.

`\texwidth`, `\texheight` 主要正文 `body` 的宽度和高度, 不包括页眉页脚

`\oddsidemargin`, `\evensidemargin` 奇数 / 偶数页, 纸张左边缘到页 `body` 左边缘的距离减去一英寸

`\topmargin` 纸张上边缘到页眉或 `body`(无页眉时) 的距离减去一英寸

`\marginparwidth` 页边注 (`marginnote`) 的宽度

`\marginparsep` 正文边缘到页边注边缘的距离

`\headheight` 页眉高度

`\headsep` 页眉底部到页主体顶部的距离

`\footheight` 页脚高度

`\footskip` 页正文最后一行底到页脚底的距离

(建议, 总把

`\oddsidemargin` (奇数页的左边界)

`\evensidemargin` (偶数页的左边界)

`\topmargin` (从上页边到页眉的距离)

设为 0cm;

如果没用页眉, 将

`\headheight` (页眉高度),

`headsep` (页眉基线到正文顶部的距离)

设为 0cm;

如果没用页脚, 将

`\footheight` (已过时 latex2e 中已被去掉),

`\footskip` (正文底部到页脚底部的距离) 设为 0cm,

这样便于在输出时直接利用驱动程序的功能控制版心的位置.)

`\columnsep` 双栏 article 中, 左右栏间空白的距离

`\columnseprule` 双栏 article 中, 左右栏间分隔线的宽度, 缺省 为零, 即没有线

以下参数的修改可以在 `\begin{document}` 的前面和后面:

`\parskip` 段落之间除了空出正常行距外额外空出的距离

`\parindent` 段首空白的长度

`\footnotesep` 两个注脚之间的距离

`\baselineskip` 正文中前一行底和第二行底的距离, 会因字号的改变自动改变.

另外, 定义行距还有一个特殊的参数, 那是 `\baselinestretch`,

它是用在 `\begin{document}` 前的全局变量, 修改格式如

`\renewcommand\baselinestretch{倍数}` 缺省时为 1.0, 因此

LaTeX 中最小行距是 `\baselineskip` 乘上 `\baselinestretch` 得到的.

+-----+

| 文档标题、摘要 |

+-----+

`\title{标题可有 \\ 换行}`

`\author{作者名可用 and 分开}`

`\date{日期可选, 无命令自动当天日期, 空白选项不显示}`

`\thanks{}` 可出现 `\maketitle` 以上任何位置, 或几个 `\thanks`

`\maketitle`

LaTeX 用特定字体、号居中输出

自定义标题环境

`\begin{titlepage}`

`\end{titlepage}`

生成标题独占一页, 并重置页码计数器

摘要环境

`\begin{abstract}`

...

`\end{abstract}`

book 类没有摘要

摘要标题中文化

`\renewcommand{abstractname}{摘 \quad 要}`

```
+-----+
| 本节内容: 三种数学环境, 数学环境中字体和字号, 数 |
| 学公式中的空格, 上下标, 分式, \displaystyle, 大 |
| 大小小的括号, 根式, 导数. |
+-----+
```

导言区加入数学公式宏包

`\usepackage{latexsym}`

`\usepackage{bm}`

`\usepackage{amsmsth}`

`\usepackage{amssymb}`

也可以等

`\usepackage{latexsym,bm,amsmsth,amssymb}`

到这里, 大家就已经能输入一篇完整的一般文章了, 但是科技文章总少不了数学公式, 而且大家选择 TeX 作为排版工具, 其中一个原因也是看中了它美观的公式编排, 现在我们就来看看和一般正文不一样的公式.

TeX 提供三种有关数学公式的环境, 一种是 "文中公式", 通过

\$ 或 \ (进入 "数学环境", 通过 \$ 或 \ 来结束数学环境的; 第二种是 "独立公式", 用 \$\$ 或 \[进入 "显示数学环境"(以后我们统称数学环境), 用 \$\$ 或 \] 退出, 在正文, 它是独占一行的, 它排出的字体要稍微比文中公式要大些; 第三种是带公式标号的数学公式: "方程", 这个以后我们详细再讲. 数学环境是自动根据当前字号来调整公式中字的大小的.

数学环境中, 缺省字体是数学斜体 (显示出来和意大利体一样), 因此如果要用到正文, 需要改变字体或使用盒子, 如我们要输出 "y=x, if x>0," 这一句, 其中的 "y=0" 和 "x>0" 应该是数学公式 (虽然不进入数学环境也能输), 而 "if" 却属于正文内容, 因此不应该用数学斜体显示, 而应该用 "\$y=x, {\rm if} x>0\$" 或用盒子: "\$y=x, \mbox{if} x>0\$", 但这里还要注意的, 数学环境中忽略一切空格 (当然分隔命令的空格除外), 因此第一种输入可以发现, if 和 x>0 之间没有我们想要的空格, 而第二种输入方式由于在 \mbox{} 中 if 后加了一个空格, 所以 if 和 x>0 有一个空格, 这正是我们想要的. 因此标准的输入方法一般用后者. 而且, 前者的方法对汉字就不适用, 因为汉字不能出现在数学环境中, 而用盒子框起来就没问题了. 这里再多说一句的是, 数学环境中不能使用字号命令, 如果需要改变字号, 应该在数学环境之外改变, 而数学环境自动根据当前字号调整字的大小.

上面已经提到数学环境中忽略任何空白字符, 那么, 怎么在公式中输出空格呢? 当然我们可以用以前讲过的 \quad 或 \hspace 来空出横向空格等, 但是如果我们要输出诸如 "fdx"(这里 dx 是微分),

这样在 f 和 dx 中要有一个小空格, 用 `\quad` 显然太大了, `hskip` 根本不知道空出多大距离. 在公式中, 常用的几个调整距离的命令是 `"\,"` (小距离), `"\:"` (中距离), `"\;"` (大距离), `"\!"` (负距离), 象刚才说的例子, 我们一般就用 $f\,dx$ 即可. 当然, 一般情况我们很少改变数学公式中的距离.

下面我们看看如何在数学公式中输入上下标, 在讲 TeX 保留字时大家就有印象了, 上下标分别用 `"^"` 和 `"_"` 来表示, 如 x_0^2 , TeX 中, 先输上标后输下标和先写下标再写上标是没有区别的, 它都会一样地自动调整字符大小和位置. 如果上下标的字符不止一个, 那就不用 `"{}"` 括起来, 如 $x^{1+y^{i+1}}$. 如果有人突然想在字符的左边输上下标呢? 简单的方法是在前面加一个空的 `{}`, 如 $\{{}_{17}^{35}\text{Cl}$, 哦, 可能不是用的数学斜体, 那只好再用盒子了: $\{{}_{17}^{35}\text{\mbox{Cl}}$.

有时我们可能直接用 $1/2$ 来表示分数就可以了, 但是如果分子分母我们想要排在分数线的上下, 那就要用 `"\frac{分子}{分母}"` 来表示了, 如
$$\frac{1}{x+1}$$
. 但是大家如果再把这个例子用单 `$` 括起做文中公式, 或者 $y=\frac{y-\frac{1}{x+1}}{y+1}$, 就会发现, 这个分式显示出来觉得比正常字号小, 要解决这个问题就要用 `\displaystyle` 的命令了. 你可以试试

$$y=\frac{y-\displaystyle\frac{1}{x+1}}{y+1}$$
, 可以了吧.

当然, 如果你在 `\documentstyle` 中使用了 `vatola` 选项 (需要 TeX 系统在指定目录有 `vatola.sty` 文件), 那么, 你可以直接用 `\dfrac` 来代替 `\frac` 来达到同样的效果, 而避免使用冗长的 `\displaystyle`

的输入.(格式如 `\documentstyle[vatola]{article}`; 而如果用 `\documentclass{article}`, 则用 `\usepackage{vatola}`); 或者二者均可以在第一句后面加上 `\input vatola.sty` 来包含这个包)

有人可能又要问了, 如果我输出的分数, 分子还是有分子分母的分数, 分母也是一个不简单的分数, 那层层迭迭的分数线, 不够美观, 我还是喜欢 "分子 / 分母" 的格式, 那怎么把这条斜分数线画得长一些呢? 这里就要用到所谓的 "定界符", 也就是成对出现的类似于括号的符号了, 如 `()`, `[]`, `{}`(用 `\{`和 `\}` 输入), `<>`(用 `\langle` 和 `\rangle` 输入), `/`, `\`(用 `\backslash`), `|`, `||`(用 `\|`); 以及另一些没法直接显示的符号, 大家自己用 TeX 产生出来看看:

`\lfloor`, `\rfloor`; `\lceil`, `\rceil`; `\uparrow`; `\Uparrow`;
`\downarrow`; `\Downarrow`; `\updownarrow`; `\Updownarrow`\$. 这些 "括号" 通常嵌套并和分数等数学公式一起使用, 如

`$$y=1+(\frac{(x+1)^2}{x^2})$$`, 当然大家会看到这外层括号太小, 不美观, 因此 TeX 提供了几个改变大小的命令, 从小到大依次是 `\big`, `\Big`, `\bigg`, `\Bigg`, 如 `$(\big(\Big(\bigg(\Bigg(`, `\Bigg)$$`, 其它定界符类似使用. 另外, TeX 还提供了自动调整括号大小的命令, 那就是在定界符前加 `\left` 和 `\right`, 需要注意的是 `\left` 和 `\right` 一定要成对出现, 如果只用单个, 那么另一边就要用 `"\left."` 或 `"\right."` 补上. 知道这些后, 我们就可以来输入长的 `"/"` 号了: `$$\frac{1}{x+1} \left/ \frac{y+1}{x^2} \right. $$` 或 `$$\left. \frac{1}{x+1} \right/ \frac{y+1}{x^2} $$`, 这两个是不是比 `$$\frac{\displaystyle\frac{1}{x+1}}{\displaystyle\frac{y+1}{x^2}}$$` 更令人满意啊?

下面, 我们来讲讲根式的输入: "`\sqrt[开方次数]{表达式}`",
其中 [开方次数] 可选参数, 如果没有则是开平方, 如
`$$\sqrt[3]{x+y}+\sqrt{2}$$`.

而导数, 也就是在字母右上方加一撇或两撇的输出, 我们常用
`$$f^{\prime}$$` 和 `$$f^{\prime\prime}$$` 来表示, 或更简单地, 用
`$$f'$$` 和 `$$f''$$` 来表示.

```
+-----+
| 本节内容: 数学符号: 希腊字母, 二元运算符, 关系运 |
| 算符, 箭头, 其它符号, 可用于一般环境的符号, 花体 |
| 符号及函数名, 可变大小的符号, \limits 和 \ nolimits, |
| 省略号, 符号的上下堆积, 上下标的分行          |
+-----+
```

TeX 提供了许多 "控制符号" 来得到诸如希腊字母, 关系运算符,
箭头等无法用 ASCII 键盘直接输入的符号. 小写希腊字母, 如 `\alpha`,
`\beta`, `\gamma`, `\pi`; 大写希腊字母, 如 `\Gamma`, `\Pi`, `\Omega`;
二元运算符, 如 `\times`(乘号), `\div`(除号), `\pm`(加減号), `\circ`
(小圆圈), `\cdot`(点) 等; 关系运算符, 如 `\leq`(小于等于), `\geq`
(大于等于), `\subset`(包含于), `\supset`(包含), `\in`(属于); 否
定关系运算符, 如 `\not=`(不等于), `\not<`(不小于), `\not\supset`
(不包含); 箭头, `\leftarrow`, `\rightarrow`, `\longrightarrow`,
`\uparrow`, `\mapsto` 等; 其它符号, `\nabla`(Nabla 算子), `\angle`(角),
`\infty`(无穷), `\forall`(任意), `\exists`(存在), `\prime`(导数的

撇)。这些符号比较多, 这里就不细细一一列出, 大家去找一本 TeX 书, 复印那两三页下来就可以了。

本节讲的绝大多数符号都只能在数学环境中使用, 有几个符号也可以使用在一般环境中, 那是 `\S`(节号), `\P`(段落号), `\dag`(剑号), `\ddag`(双剑号)。

有时数学公式中的函数名, 算子等要用特别的花体来排版, TeX 提供了 `\cal` 字体, 如 "function $\cal F$ ". 而对于专有名词, 如一些函数名, 如 $\sin x$ 中的 \sin , 就要用罗马体, 而不是一般的数学斜体排印, 我们可以用 $\rm \sin x$, 也可以用 TeX 提供的直接在函数名前加 `"\"` 的方法: $\sin x$, 一般的函数均有定义, 如 `\sin`, `\cos`, `\lim`, `\log` 等。

数学环境中, 还有一类特殊的符号, 那就是可变大小符号, 如求和号, 积分号, 它们会因为符号后面的内容的大小自动调整自己的大小: `\sum`, `\prod`, `\coprod`, `\int`, `\oint`, `\bigcap`, `\bigcup`, `\bigsqcup`, `\bigvee`, `\bigwedge`, `\bigodot`, `\bigotimes`, `\bigoplus`, `\biguplus`. 需要注意的是, 有些可变符号的上下标位置在文中公式和独立公式中是不一样的, 大家看个例子:

$\sum_{i=1}^n x_i = \int_0^{\infty} f(x) dx$

和 $\sum_{i=1}^n x_i = \int_0^{\infty} f(x) dx$, 这里大家看到,

`\sum` 在文中公式中, 上下标是写在右上角和右下角的, 在独立公式中, 是写在上面和下面的; 而 `\int` 在二者中都写在右上角和右下角的. 我们可以强制用 `\limits` 和 `\nolimits` 来控制上下标的出现位置:

使用方法如 $\sum\limits_{i=1}^b x_i$ 就强制将上下标写在上边和下边, 而 \nolimits 使其出现在右边角上. 这里多说一句, 如果你想把一般的操作符也想用 \limits 或 \nolimits 用, 只须把你的操作符用 $\mathop{\}$ 括起来即可.

大家如果编排矩阵或行列式, 就会遇到特别的省略号, 大家试试 \cdots , \vdots , \ddots 就是三个方向的省略号. 而 \ldots 是比 \cdots 低一点的省略号, 它和其它三个不同的是也可以用在正文环境中. \ldots 一般用于 x_1, x_2, \ldots, x_n 这样的時候.

排版数学公式可能还会遇到需要把两个符号上下组合在一起, 最常见的就是上划线和下划线, 大家试试 $\overline{a+b+c+d}$ 和 $\underline{a+b+c+d}$, 这里需要说明的是, $\overline{\hspace{1cm}}$ 只能用于数学环境, 而 $\underline{\hspace{1cm}}$ 也可以用在正文中. 类似地还有用花括号括在式子上下的: $\overbrace{a+\underbrace{b+c}+d}$, 而如果上下括号上面或下面还有符号, 可以用上下标来表示, 如: $\overbrace{a+\underbrace{b+c}_{1.0}+d}^{2.0}$. LaTeX 还可以将任意两个符号重叠在一起: $\stackrel{\text{上层符号}}{\text{下层符号}}$, 如 $\vec{x} \stackrel{\text{rm def}}{=}(x_1, x_2, \ldots, x_n)$, 这里需要说明的是 $\stackrel{\text{rm def}}{=}$ 中第一个参数的符号以较小字号输出的. 而 $\overset{\text{上面}}{\underset{\text{下面}}{\hspace{1cm}}}$ 可以把两行内容按相同字号排出, 类似地就出现了 $\binom{n}{m}$ 则是带括号的 n 个取 m 个的写法.

符号可以上下堆积, 可是我们如果要在 \sum 的下面写诸如 $i < 9$ 和 $j < 9$ 这两行下标, 用 $\sum_{i < 9 \setminus j < 9} i/j$ 是不行的. 我们

怎么实现这样的分行呢? 我们可以用 `\atop` 来堆积, 如

$$\sum_{\scriptstyle i<3 \atop \scriptstyle j<3} i/j$$

这里之所以加了 `\scriptstyle` 是因为仅用 `\atop` 命令会把上下的字

体再缩小一号. 另外, 我们也可以用另外的方法: 前面我们在讲

`\frac` 时提到 `\dfrac` 这个符号需要使用包 `vatola.sty`, 这里, 如果

使用了 `vatola.sty` 这个包, 我们可以在下标中用 `\Sb` 第一行 `\`

第二行 `\endSb`, 在上标中用 `\Sp` 第一行 `\` 第二行 `\endSp` 来

表示多行下标或上标.

```
+-----+
| 本节内容: 数组 array, 方程 (标号)equation, 多行方程 |
| eqnarray, \aligned(vatola.sty), 标号的交叉引用    |
| 参考文献的使用 (more)                               |
+-----+
```

数组环境通常用来编排矩阵, 行列式等对齐的数学公式的.

它的格式是:

`\begin{array}{列对齐}`

第一行 `\`

第二行 `\`

...

最后一行

`\end{array}`

这里, 列对齐的格式是 `{c1c2c3...cn}`, 每个 `ci` 为一个字母, 说明

第 `i` 列的对齐方式, 可以是 `c`(居中), `l`(左对齐), `r`(右对齐). 而

每一行的各列用 & 符号隔开, 行末是 \\. 因此 n 列的数组, {列对齐}
应该有 n 项, 每行应该有 n-1 个 &, 一个 \ (最后行通常不用 \). 数组中某一列可以不存在, 但是 & 号不能少. 数组外常常使用可变大小的定界符 \left 和 \right, 大家看个例子:

```
$$\left(  
\begin{array}{clr}  
a+b+c & a+b+c & a+b+c \\  
a+b & a+b & a+b \\  
a & a & a  
\end{array}  
\right) $$
```

这里要作些说明, 数组中某两行的距离可以用 "[距离]" 来改变, 也可以用 "\renewcommand\arraystretch{实数}" 来整体修改数组的垂直距离 (缺省为 1.0), 而参数 \arraycolsep=... 是修改其列间距. 使用数组时, 经常容易出错的是: array 必须出现在数学环境中, & 的个数和 \ 的个数一定要正确, 如果使用可变大小的括号, 一定要注意 \left 和 \right 必须成对出现, 即使只排印半对括号.

一般的 \$ 和 \$\$ 环境, 不能给公式或方程编号, 因此, 就用到了方程环境. 格式是:

```
\begin{equation}  
方程  
\end{equation}
```

这其中的方程, 和 $\$$ 情况没有两样, 只是在公式右边自动加上了方程编号 (如果 `\documentstyle` 用了 `\leqno` 选项, 则公式标号在左边). 多行的方程可以把 `array` 使用在 `equation` 中来实现, 但是标号只出现一个, 出现在多行的中间. 如果多行要分别标号, 或者不想用麻烦的 $\$$ 嵌套 `array`, 那就要用到多行方程环境:

```
\begin{eqnarray}
```

```
.....
```

```
\end{eqnarray}
```

和

```
\begin{eqnarray*}
```

```
...
```

```
\end{eqnarray*}
```

二者的区别是, 带 `*` 号的不将公式标号排出来, 而不带 `*` 的自动给每行式子编排标号. 在 `\begin{eqnarray}` 和 `\end{eqnarray}` 中的部分是一个三列的 `array` 环境, 因此有两个 `&` 和一个 `\\`, 如:

```
\begin{eqnarray}
```

```
y&=&1+2+3+4+5+6+7+8+9+10 \\
```

```
&=&55
```

```
\end{eqnarray}
```

而如果把方程某行的 `\\` 前加上 `"\nonumber"`, 则 TeX 不给该行方程编排标号, 如 `"y&=&1+2+3+4+5+6+7+8+9+10 \nonumber \\"`, 则该行不出现标号. 这里还要注意的是 `equation` 和 `eqnarray` 都是直接进入 $\$$ 的显示数学环境, 因此不能在前面和后面加上 `$` 或 `$$` 了.

大家可能会注意到, 不论是 `array` 还是 `eqnarray`, 每个 `&` 符号

都会在公式中产生列的小空隙, 同样也在 `vatola` 宏库中, 定义了一个 `aligned` 对齐环境, 它使得编排多行公式在对齐的同时不会出现 $\$$ 符号位置的列空隙. 它可以出现在 $\$ \$$ 环境中, 那公式不标号; 或出现在 `equation` 中, 它对一个多行公式只标一个标号, 类似于 `array`. 它相当于一个两列的 `array`, 因此每行仅有一个 `&`.

```
\begin{equation}
```

```
\aligned
```

```
y&=1+2+3+4+5+6+7+8+9+10 \\\
```

```
&=55
```

```
\endaligned
```

```
\end{equation}
```

这里需要注意的是, `\begin{array}` `\end{array}` 和 `\begin{tabular}` (制表) `\end{tabular}` 不能出现在 `\aligned` 与 `\endaligned` 中, 如须使用, 要用 `\vbox{}` 或 `\hbox{}` 等把 `array` 或 `tabular` 环境括起来.

在讲第一个完整的 TeX 文档例子时, 大家一定注意到参考文献标号的交叉引用, 而方程也有标号, 也可以交叉引用. 为什么我们选择交叉引用而不是直接在文中写 "公式 (1) 中..." 的原因很简单, 交叉引用可以实时地调整标号, 插入或删除一个公式带来整体标号的改变, 无需改变引用标号的地方; 而且, 可以在不修改正文的情况下灵活地调整标号的风格.

TeX 使用 `\label{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 `\label` 定义过的, 定义和引用的先后无关.

例如, 我们有

```
\subsection{Early Results}
```

```
\label{sec-early} % 这里定义子节标号
```

Euler's equation

```
\begin{equation}
```

```
e^{i\pi}+1=0 \label{eq:euler} % 这里定义方程标号
```

```
\end{equation}
```

\combines the five most important numbers in mathematics
in a single euqtion.

然后我们就可以如下地引用

Equation~\ref{eq:euler} in Section~\ref{sec-early} ...

不管这里方程或子节的标号是多少, 引用时都能准确地指出. 这
就是我们要使用自动交叉引用的原因. 这里, \label 会自动根据
自己的位置, 得出标号, 如出现在 equation, eqnarray, enumerate
(项目), figure(图), table(表) 环境中, 被引用的是公式号, 项
目编号或图表号; 而出现在其它位置则是章节号.

引用时除了 \ref 外还有一个 \pageref{标号}, 它显示的是
\label{标号} 出现的页面位置.

\cite 和 \bibitem 我们在一开始的例子中就看到了, 我们这里
更详细地说说这两条命令的完整格式:

```
\cite[附加信息]{标号 1, 标号 2, ..., 标号 n}
```

```
\bibitem[名称]{标号}
```

先看一个 \cite 加附加信息的例子 (这个例子是在第 5 节例子的参考

文献的基础上的):

See \cite{texbook, companion} or \cite[page 20–22]{lamport} ..

再看 \bibitem 的名称可选项:

See \cite{kn:texbook} for more details.

...

\begin{thebibliography}[lamport 86] %[] 中是最长文献编号

\bibitem[Knuth 84]{kn:texbook} D. E. Knuth. {\sl The \TeX{}book}.

Addison–Wesley, Reading.

\end{thebibliography}

这里再多说一句的是: 所有涉及交叉引用的文件, 在修改过与交叉引用有关的内容之后, 都需要对源文件进行两次编译才能得到正确的结果.

```
+-----+
| 本节内容: 居中, 左对齐, 右对齐环境; 小页环境; 抄 |
| 录环境; 列表环境; 制表环境. |
+-----+
```

这里要介绍的是几个 TeX 中常用的环境, 它的一般格式是:

\begin{环境名}

...

\end{环境名}

首先是居中, 左对齐, 右对齐环境, 这三者的环境名分别是 center, flushleft 和 flushright, 在环境中的正文以 \\ 断行. 这

三个环境比较简单, 就不举例子了. 需要注意的是, 紧接这环境后的正文和环境如果没有空行隔开, 那么系统认为是一个段落, 就是说环境后的正文行首没有 `\parindent` 的缩进.

小页环境大家在第一个完整的 TeX 例子中已经看到了, `abstract` 就是小页. 小页环境的格式是:

```
\begin{minipage}[位置]{宽度}
```

...

```
\end{minipage}
```

`{宽度}` 可以用 TeX 的任何合法距离; `[位置]` 是可选项, 可以选择 `t` 或 `b` 来表示小页是和正文是 "顶部对齐" 还是 "底部对齐", 请看例子:

NORMAL TEXT

```
\begin{minipage}[b]{1 in}
```

This is a minipage aligned on its bottom line.

```
\end{minipage}
```

AND

```
\begin{minipage}[t]{1 in}
```

This is a minipage aligned on its top line.

```
\end{minipage}
```

END

注意的是, 小页的 `footnote`(脚注) 是紧跟在小页后面, 而不是在整个页面的下面, 所以在 `minipage` 嵌套时, 可能系统会放错脚注的位置.

抄录环境 (`verbatim`) 在 `\begin{verbatim}`和 `\end{verbatim}`

的任何字符都将原样输出, 包括 \ 等 TeX 保留字. 而 "verbatim*" 与 verbatim 的区别是, 它将把空格用 | 表示出来. verbatim 有一个简写形式, "\verb 标识符 字符串 标识符", 其中标识符可以是空格以外任何字符, 它与 \ verb 之间没有空格. 如:

```
\verb+
```

```
This is \!@#$$%%^&*&*({ All you want to type.
```

```
+
```

当然, 字符串里就不能出现你作为标识符的符号了.

列表环境 itemize, enumerate, description.

```
\begin{itemize}
```

```
\item 列表项 1
```

```
\item 列表项 2
```

```
...
```

```
\item 列表项 n
```

```
\end{itemize}
```

自动编号; 如果超过一行, 自动缩进以突出编号. 而 enumerate 与 itemize 的区别是, 它不出现数字编号, 而是以小黑圆点来标识.

description 一般用于类似于名词解释的情形:

```
\begin{description}
```

```
\item[名词] 解释
```

```
\item[名词] 解释
```

```
...
```

```
\end{description}
```

一般, 还在 [名词] 中加上 \bf 或 \heiti 以突出名词.

列表环境还可以嵌套使用, 具体就不细述了.

```
+-----+
```

```
| 表 格 |
```

```
+-----+
```

表格环境 tabular

```
\begin{tabular}[竖向位置]{列格式}
```

```
....
```

```
\end{tabular}
```

竖向位置: t、b 缺省居中

列与列间用 & 分隔, 每行 \\[长度: 改变行间间隔] 结束

列格式: l 左对齐, c 居中, r 右对齐

边界线: |, ||

横线:

```
\hline
```

```
\cline{m--n} m--n 列水平线
```

```
\nline 与行等高竖线
```

```
\multicolumn{列数}{列格式}{文本内容}
```

列数 = 1 时, 改变当前列对齐方式

TeX 专门提供了一个制表环境, 用来排印有线或无线的表格.

```
\begin{tabular}[位置]{列定义}
```

```
...
```

```
\end{tabular}
```

这里 [位置] 和 minipage 一样, 可选 t 或 b, 但我们通常用的表格都是独立表格, 所以通常不定义位置, 而直接独立一段. {列定义} 类似于 array 的列对齐, 可以用 c,l 和 r 来表示对齐方式, 这里的列定义还可以在列对齐字母间用 "|" 以画出表格中列的竖线. 表格内容如 array 一样, 用 & 隔开, \\ 换行, 同样要注意 & 的个数. 表格的横线可以用 \hline 来表示, 如:

```
\begin{tabular}{|c|c|c|}
```

```
\hline
```

```
1 & 2 & 3 \\
```

```
\hline
```

```
4 & & 6 \\ % 这里要注意, 若最后一行要画横线, 则此处要有 \\
```

```
\hline
```

```
\end{tabular}
```

而 "\cline{列号 1 – 列号 2}" 则是表示从列号 1 到列号 2 的一条横线.

而如果要在表格元素中画竖线, 则可以用 "\vline". 有时我们还会遇到需要把某行表格的几列合并起来, 这就要使用

```
\multicolumn{栏数}{栏定义}{内容}
```

{栏数} 是要合并的栏的数目, {栏定义} 类似于表格的 {列定义}, 如:

```
\begin{tabular}{|c|c|c|}
```

```
\hline
```

```
\multicolumn{2}{|c|}{1} & 2 \\
```

```
\hline
```

```
3 & 4 & 5 \\
```

```
\hline
```

```
\end{tabular}
```

而

```
\begin{tabular*}{宽度}[位置]{列定义}
```

...

`\end{tabular*}` 可以自己定义表格的总宽度, 如 `\textwidth` 等 TeX 的合法距离.

最后, 我们讲讲制表环境的四个参数, 这四个参数都是局部参数, 也就是说只要用分组限定命令的作用范围, 就可以调整某个表格的参数而不影响其它表格. `\tabcolsep` 是两列间水平距离的一半, 缺省为 6pt; `\arrayrulewidth` 为定义 `\hline`, `\vline`, `\cline` 及列定义的分隔线 | 的线宽, 缺省为 0.4pt; `\doublerulesep` 为连续两个 `\hline` 或列定义中连续两个 | 所画的线段之间的间隔, 缺省为 2pt, 如果设为 0pt, 则可以用连续的 `\hline` 或 | 来加粗表格的某些线框; `\arraystretch` 是一个实数, 缺省为 1.0, 定义行距的一个因子, 用 `\renewcommand` 来改变.

```
+-----+
| 本节内容: 在 TeX 文档中插入 eps 图形文件 |
+-----+
```

plain TeX 的图形功能非常弱, LaTeX 在一定程度上增强了这方面的功能, 但是也只能画由文字, 直线, 矢量, 圆等组成的很简单的图形. 这里我觉得没有多大必要花不少精力去学这不大实用的东西. CCT 对图形, 图像功能方面作了补充, 具体大家可看它

所附带的 man.dvi 文件, 里面有详细的说明. 而 windows 下的 TeX 或如 sci-word 一般的环境, 可能提供了插入一般 windows 下剪贴板图形的方法, 具体可看该软件的帮助.

这里, 我们仅就如何在 TeX 文档中插入 EPS 的 PostScript 图形作些说明, 因为现在的很多 windows 下的作图软件, 如 mathematica 都能产生 EPS 文件, 这类文件的图形是矢量格式而不是点阵格式的, 可以以较高精度输出而不会出现锯齿状不光滑现象.(mathematica 3.0 中, 只要点击所做的图形成为选中, 然后点鼠标右键, 在菜单中选择 Save Selection As..., 那里有 EPS 格式)

插入 EPS 图形的 TeX 文件所生成的 dvi 文件, 无法用 view 看到其中嵌入的图形, 只能用 dvips 将 dvi 以及内嵌的 EPS 图形一起转化成 PS 文件, 然后用 Ghost View 来看.

首先, 我们先要安装 "dvips". 现在假设我们已经把 CCT EMTEX 5.13 安装在 C:\EMTEX 目录下了 (如果装在其他盘, 那么 dvips 就要改设置 config.ps 了, 估计有麻烦), 那么解开 dvips.zip(下载到 <http://202.38.68.78/~chencs/math/>), 里面有一部分文件展开在 emtex 目录下, 还有一个 texfonts 目录 (大概是为 dvips 提供字体用的). 大家使用 CCT 之前, 应该运行过 emtex 目录下的 setenv.bat (这是 CCT emtex 要求大家设置的 TeX 的环境变量, 建议将 setenv.bat 的内容加到 autoexec.bat 中去, 因为这里的环境变量没设好, 会影响 dvips 的使用).

有了上面准备工作之后, 以后如果我们要在 TeX 中插入 EPS 文件, 只要按 TeX 格式编好源程序; 然后用 patchdvi 把 dvi 文件中可能有的汉字字库剥离出来, 以供不依赖汉字环境的 dvips 使用; 然后用 dvips32 就可以将 dvi 文件以及 EPS 图形一起合成为 ps 文件, 回到 95 环境, 用 GhostView 就可以阅读或打印了.

在打包的 dvips.zip 中, 我们加入了一个 ll.bat 文件 (使用格式是 ll filename, 这里, CCT 的文件名为 filename.ctx), 大家可以从中看到怎么使用 patchdvi 和 dvips:

```
call patchdvi -y %1.dvi ~~tmp.dvi
dvidrv dvips32 ~~tmp.dvi
```

最后, 大家就看看怎么在 TeX 文件插入 EPS 图形了. 如果我们用 `\documentstyle{}`, 那么, 在 `{}` 前的选项 `[]` 中加入 `epsfig`; 如果用的是 `\documentclass{}`, 那么就用 `\usepackage{epsfig}` 调如该库. 然后在要插入图形的地方使用 `\epsfig{figure=filename.eps}`, 当然, 也可以把这句话加在浮动图表的环境中而形成浮动图表, 或直接使用直接插在文中. 一个简单的例子是打包在我们的 dvips.zip 中的 sample.ctx 中插入了 sin.eps.

这里补充说一下浮动图形 figure 环境, 它能自动调整图形在页面中出现的位置:

这里 `[位置]` 可以是 `h`(当前位置), `t`(页顶), `b`(页底), `p`(另页).

+-----+

| 一个英文文档的例子 |

+-----+

一个英文文档的例子，保存为:*.tex

`\documentstyle{article}` % 使用 article 格式

`\topmargin=0pt` % 以下页面设置

`\oddsidemargin=0pt \evensidemargin=0pt` %

`\textwidth=14cm` %

`\textheight=21cm` %

%

`\title{This is a sample of \LaTeX}` % 文章题目

`\author{Bigeyes(\tt{chencs@263.net})\\}` % 作者. 多行用 \\

Department of Mathematics\\ %

USTC, Anhui, Hefei, 230026 %

%

`\and` % 如有多作者, 用 \ and

`A_A` %

`\thanks{Supported by ...}` % 致谢, 在 author 里

\\ %

BBS of USTC\\ %

bbs.ustc.edu.cn[202.38.64.3] %

`\date{Mar. 1, 1999}` % 日期, 如果没有此项,

% 则以当前日期代替, 若

% 不要日期, 则用空 {}

%

%document 结束

```
+-----+  
| 如何编辑、编译 TeX 文件 |  
+-----+
```

我们已经讲过, TeX 源文件是文本文件, 就象一个 C 语言的源文件一样, 那么我们也象编译 C 语言程序得到可执行程序一样, 我们编译 TeX 源文件, 以得到能在计算机上浏览或打印的格式文件, 那就是后缀为. DVI 的文件 (我们就称它为 DVI 文件). TeX 源文件我们通常把其后缀取为. TEX. 当然, 如果你在中文 TeX 的环境中编排你的中文文章, 那么你的源文件后缀可能就是那个中文 TeX 给你约定的了, 我们现在常用的 CCT EMTEX, 所用的后缀是. CTX, 系统的操作是将. CTX 源文件中的中文经过预处理, 产生标准的 TeX 文件, 然后再照平常的 TeX 文档一样操作, 得到 DVI 文件.

今后, 我们均以 DOS 下的 CCT EMTEX 环境为例, 来讲述 TeX 文章的编排. 因为 TeX 的标准, 所以在不同的 TeX 环境, 其实文章编排的内容格式是一样的, 只是可能它得到 DVI 的途径不同而已. 而我们就以现在比较流行的 LaTeX 为准来讲述 TeX 的使用, 其它如 amstex, amslatex 等, 仅是文档的最外层框架以及少数扩展的命令不大一样之外, 其它命令都是相同的.

一个最简单的 TeX 文档可能是: (比如我们用 EDIT, 当然也可以用写字板或 UltraEdit 等甚至是 UNIX 下的 vi 编辑器等编辑一个文本文件如下四行, 取名为'1.tex')

```
\documentstyle{article}
```

现在流行的 TeX 格式似乎已经不是 documentstyle 而是 documentclass 了, 但是它们除了一开始的写法不大一样之外, 正文的命令排版等等都没有任何区别. 以 CCT EMTEX 环境为例, 我们用'latex 1.tex'来编译 TeX 文件来得到 DVI 文件, 然后用'view 1.dvi'来浏览, 用'dvi24p 1.dvi'(24 针打) 或'dviljp 1.dvi'(激打) 来打印文章. (以上三条命令的后缀均可以省去, 仅用'latex 1','view 1'和'dviljp 1'等.)

如果你的 TeX 文件不符合 TeX 语法, 那编译将可能出错, 这时可以让系统忽略错误产生 DVI 文件 (这时可能就不能达到你所期望的排版效果了), 或者你应该修改源文件中的错误, 重新编译, 直到全部通过. 在 LaTeX 编译出错时, 会在屏幕上显示出错信息, 并显示 "?" 以接受用户进一步的指示. 在 "?" 后, 输入:

1. 回车: LaTeX 尽量忽略此错误, 继续运行;
2. s: LaTeX 继续运行, 并之后还有错误, 也不停止运行;
3. r: 类似 s, 比上者更强硬, 甚至连打不开文件这样的严重错误也不停止;
4. q: 同 r, 但错误及提示均不在屏幕显示, 直接写入 LOG 文件;
5. l: 之后跟一个字符串, 用来插入替代出错的字符串. 比如在

正文需要纵向空出 1cm 空白, 用 "\vskip 1cm", 但是输入错误

成为 "`\vship 1cm`", 则编译出错:

! Undefined control sequence.

l.102 `\vship`

1cm

?

表示在第 102 行, `\vship` 是未定义的命令, 这时在? 后输入 "`\vskip`", 则用 `\vskip` 替代出错的 `\vship`, 继续执行. 这里需要注意的是, 这条指令只是编译中临时修改错误, 并没有真正地修改源文件;

6. 小于 10 的数: 表示删出从当前位置开始的那么多个字符;
7. ?: 显示提示
8. h: 显示帮助, 对当前错误提出进一步动作的建议;
9. x: 中断编译, 将已编译的结果写入. DVI 中, 出错及提示写入 .LOG 中.

+-----+

| 制作幻灯片 |

+-----+

xcolor 宏包

`\usepackage[选项]{xcolor}`

选项:

彩色驱动程序名: dvips, dvipdfm, pdftex

彩色模式: natural, rgb, cmyk, ..

预定义色彩集合: dvipsnames, dvipsnames*, ...

red,green,blue,yellow,cyan,magenta,black,white,orange,violet,purple,brown,pink,...

使用彩色

`\color{色彩名}`--> 声明

`\textcolor{色彩名}{文本}`--> 命令

`\pagecolor{色彩名}`--> 声明，指定背景色（ps，pdf 有效）

色彩名必须是已经有定义的

可以使用不同色彩的混合，如

`\color{green!40!yellow}`-->40% 的绿色与 60% 的黄色混合

`\color{-green!40!yellow}` 上述颜色的补色

色彩混合公式：

$C_0!P_1!C_1!P_2!...!P_N!C_N=????$

另一种方法

`\color[色彩模式]{数据}`

`\textcolor{色彩名}{数据}{文本}`

`\pagecolor{色彩名}{数据}`

例：

`\textcolor[rgb]{0.3,0.8,0.1}`

`\textcolor[cmYk]{0.0,0.5,0.1,0.5}`

`\definecolor{色彩名}{色彩模式}{数据}`

`\providecolor{色彩名}{色彩模式}{数据}`

色彩模式：rgb，cmYk，gray 等

逗号分隔的十进制小数，表示每个分量的力度

如：`\definecolor{red}{rgb}{1,0,0}`

```
\definecolor{yellow}{cmyk}{0,0,1,0}
```

如色彩名已有定义，\providecolor 不起作用，\definecolor 覆盖原来的定义

彩色盒子

```
\colorbox{色彩名}{文本}
```

```
\fcolorbox{色彩一}{色彩二}{文本}
```

```
\colorbox{色彩模式}{数据}{文本}
```

```
\fcolorbox{色彩模式}{数据一}{数据二}{文本}
```

\normalcolor: 恢复正常色

使用 pdfTeX

dvipdfm-->div 转 pdf

dvipdfmx-->dvipdfm 的扩展

pdfTeX 处理 Plain TeX and AMS-TeX 文件

pdfLaTeX 处理 LaTeX 文件

直接生成 pdf 文件，可以增加一些 LaTeX 中没有的 pdf 功能

支持图片格式：pdf, png, jpg, tif, 但不支持 eps

建立书签：hyperref 宏包

```
\usepackage[选项]{hyperref}
```

```
\usepackage[选项]{hyperref}
```

```
hypersetup{...}
```

例子：

```
\usepackage[dvipdfmx,CJKbookmarks]{hyperref}
```

```
\usepackage[dvipdfmx]{hyperref}
hypersetup{CJKbookmarks=true}
```

hyperref 宏包常用选项

驱动选项: dvipdfmx, dvips, pdftex,

pdf 属性: pdftitle,pdfauthor,pdfsubject,...

pdf 打开方式: pdfstartview,pdfpagemode,...

书签: bookmarks,CJKbookmarks,bookmarksnumbered,bookmarksopen,...

链接属性: colorlinks,linkcolor,citecolor,...

slides 文档类: 制作透明胶片

```
\documentclass[选项]{slides}
```

字体较大, \normalsize-->20pt

缺省字体为 sffamily

三种环境: slide,overlay,note

pdfscreen 宏包

```
\usepackage[选项]{pdfscreen}
```

常用选项

screen: 产生适合屏幕输出的 pdf 文件

print: 输出打印版

panelleft, panelright, nopanel: 设置导航面板

paneltoc: 将目录放在导航面板上

sectionbreak: 每节分页

导航面板配色方案: blue, gray, orange, ...

pdfscreen 宏包自动输入 graphicx 和 color 宏包

pdfscreen 定制幻灯片命令 (导言区)

\screensize{高度}{宽度}: 屏幕大小 (必须)

\margins{左}{右}{上}{下}: 定义页边空白, (放在 \ screensize 前)

\overlay{pdf 图形文件}: 幻灯片背景

\paneloverlay{pdf 图形文件} 导航面板背景

\urlid{CRL 地址}: 设置主页链接地址

\emblema{图形文件}: 在导航面板加入图片

导航按钮:

\bottombuttons, \nobottombuttons

\topbuttons

\notopbuttons

按钮中文化

\panelhomepagename: 主页

\paneltitlepagename: 标题页

\panelabstractname: 摘要页

\panelcontentsname: 目录页

\panelgobackname: 返回

\panelfullscreenname: 全屏显示

\panelclosename: 关闭

\panelquitname: 退出

\panelpagename, \panelofname, \panelofnamep

slide 环境

一个 slide 环境生成一张幻灯片

页面过渡命令（需使用宏包 texpower）

```
\usepackage[display]{texpower}
```

```
\pageTransitionBlindsH
```

```
\pageTransitionBlindsV
```

```
\pageTransitionSplitVI
```

```
\pageTransitionDissolve
```

```
\pageTransitionSplitVO
```

```
\pageTransitionSplitHI
```

```
\pageTransitionSplitHO
```

```
\pageTransitionGlitter{270}
```

+-----+

|beamer 宏包（最佳）|

+-----+

是 latex 制作 slide 的一个文档类

用 pdflatex 编译，不需要其它后处理程序

同票 latex 结合度高

直观而方便的创建遮挡（overlays）

及其它各种漂亮的演示效果

利用 theme 实现内容和风格的独立，可以轻松的改变风格

在 latex 中使用 beamer 的一般步骤

将 latex 的文档类型从 article 改为 beamer

用 section 和 subsection 组织 latex 的文档结构

用 frame 环境制作独立的帧

执行 pdflatex 编译 tex 文件

beamer 文档类

```
\documentclass[11pt,CJK,compress]{beamer}
```

```
\usetheme{Warsaw}
```

```
\usecolortheme{lily}
```

...

幻灯片主体结构

```
\section{....} % 用于书签和导航条
```

```
\subsection{....} % 用于书签和导航条
```

一个帧中可以包含覆盖，逐段显示内容

一个有用的选项：plain

标题页

标题，副标题，报告人，。。。

```
\title{...}
```

```
\subtitle{...}
```

```
\author{...}
```

```
\date{...}
```

```
\institute{...}
```

```
\titlegraphic{...}
```

目录页

`\tableofcontents`--> 逐段显示目录

`hideallsubsections`, `hideothersubsections`

`currentsection`

`currentsubsection`

若不想使用自动生成的目录，可以手工输入

中文支持 – CJK 宏包

中文模板

```
\documentclass[CJK]{beamer}
```

.....

```
\usepackage{CJK}
```

.....

beamer 中的一些预定义环境

`theorem`, `lemma`, `corollary`, `definition`, ...

Theorem (费马)

`block` 环境 --> 类似定理环境，可自选标题

columns 环境

beamer 中的罗列环境

应多使用罗列环境

可以嵌套使用，但通常不要超过 2 层

罗列条目的自动逐段显示

`\beamerdefaultoverlayspecification{<+>}`

手工指定逐段显示的内容：<数字范围>

如：<3>,<-3>,<2-4>,<2->

抄录环境 `semiverbatim`

使用该环境，需在 `frame` 后加 `[fragile]` 或 `[containsverbatim]` 选项

`\`，`{`，`}` 仍然保持特殊含义

`verbatim` 仍然可以使用

条目不同性质内容用 `\newblock` 分隔

插入图片

`\includegraphics[scale=0.5]{***.png}`

插入多媒体： `multimedia` 宏包

插入电影

`\movie[autostart]{***.wmv}`

插入声音

`\sound[autostart]{***.wav}`

插入 logo

`\pgfdeclareimage[height=.8cm]{logo}{math}`

`\logo{\pgfuseimage{logo}}`

在幻灯片右下角插入 logo

幻灯片的切换

`\transblindshorizontal`

`\transblindsvertical`

`\transboxin`

`\transboxout`

`\transdissolve`

`\transglitter`

`\transsplithorizontalin`

`\transsplitverticalin`

`\transsplithorizontalout`

`\transsplittverticalout`

`\transwipe`

导航图标

`\setbeamertemplate{navigation symbols}%{\insert***navigation symbol}`

其中 *** 可以为

slide,frame,subsection,section,doc,backfindforward

多个导航图标，用逗号分隔

取消

`\setbeamertemplate{navigation symbols}{}`

+-----+

|beamer 中动态时间和倒计时的最终解决办法 |

+-----+

摘自 bbs.ctex.org

pdflatex 通过。不可以用 web 宏包，否则和 beamer 冲突。

颜色字体，时间格式等参数自己设。

有些要参考 [JavaScript](#) ，如时间格式。

```
\documentclass[pdftex,8pt]{beamer}
\mode{
\usetheme{CambridgeUS}
\useinnertheme{default}
\usecolortheme{orchid}
\usecolortheme{whale}
\usefonttheme{default}
}
```

```
\usepackage{pgf,pgfrcs,pgfarrows,pgfnodes,pgfautomata,pgfheaps,pgfshade}
\usepackage{xcolor}
\usepackage{amsmath,amssymb}
\usepackage[latin1]{inputenc}
\usepackage{colortbl}
```

```
\usepackage[english]{babel}
\usepackage{times}
\usepackage{hyperref}
\usepackage{graphicx}
\usepackage[final]{pdfpages}
\usepackage{textcomp}
\usepackage{multicol}

%*****

\usepackage{eforms}
%*****


\let\oldfootnote\footnote
\renewcommand\footnote[1]{\oldfootnote{%
\renewcommand\baselinestretch{1}%
\large\scriptsize\ignorespaces#1}}
\addtolength{\footnotesep}{1pt}


\setlength{\abovecaptionskip}{2pt}
\setlength{\belowcaptionskip}{1pt}
\makeatletter
\newcommand\figcaption{\def\@captype{figure}\caption}
\newcommand\tabcaption{\def\@captype{table}\caption}
\makeatother
```

```
\setbeamertemplate{caption}[numbered]
```

```
\usepackage[absolute,overlay]{textpos}
```

```
\usepackage{tikz}
```

```
\title[Title short]{Title long}
```

```
\subtitle{Subtitle subtitle}%
```

```
\author[authorshort]{\large{authorlong}}%
```

```
\institute[Schoolshort]{\large{School of .....}}%
```

```
%*****
```

```
\newcommand{\timemark}%
```

```
{\textfield[\BC{0.2 0.2 0.7}\BG{0.2 0.2 0.7}}%
```

```
\textfont{TiRo}\textsize{5}\textcolor{1 1 1 rg}]{datetime}{2cm}{4bp}}
```

```
% 参数可以自己改
```

```
\date[\timemark]{\today}% 时间标签
```

```
%*****
```

倒计时

```
+-----+
```

```
| 例    子 1|
```

```
+-----+
```

```
% 13-4-1.tex 用 pdfLaTeX 编译
```



```
\documentclass[cjk]{beamer}
\mode<presentation>
{
  \usetheme{Warsaw}
  % 可供选择的主题参见 beameruserguide.pdf, 第 134 页起
  % 无导航条的主题: Bergen, Boadilla, Madrid, Pittsburgh, Rochester;
  % 有树形导航条的主题: Antibes, JuanLesPins, Montpellier;
  % 有目录竖条的主题: Berkeley, PaloAlto, Goettingen, Marburg, Hannover;
  % 有圆点导航条的主题: Berlin, Dresden, Darmstadt, Frankfurt, Singapore,
  Szeged;
  % 有节与小节导航条的主题: Copenhagen, Luebeck, Malmos, Warsaw

  % \setbeamercovered{transparent}
  % 如果取消上一行的注解 %, 就会使得被覆盖部分变得透明 (依稀可见)
}

\usepackage{CJK}

\begin{document}
\begin{CJK*}{GBK}{kai}
\CJKtilde
\title{没有最大的素数}

\author[\CJKfamily{fs} 王某某]{王某某 \texttt{who@where.com}}
```

```
\date[CFP 2003]{奇妙问题探讨大会, 2003}
```

```
\begin{frame}
```

```
\titlepage
```

```
\end{frame}
```

```
\begin{frame}
```

```
\frametitle{概要}
```

```
\tableofcontents
```

```
\end{frame}
```

```
\section{动机}
```

```
\subsection{我们研究的基本问题}
```

```
\begin{frame}
```

```
\frametitle{什么是素数}
```

```
\begin{dfn}
```

```
\alert{素数} 就是只有两个因子的数.
```

```
\end{dfn}
```

```
\begin{ex}
```

```
\begin{itemize}
```

```
\item 2~ 是素数。
```

```
\pause
```

```
\item 3~ 是素数。
```

```
\pause
```

```
\item 4~ 不是素数~(它有~\alert{3}~ 个因子)。
\end{itemize}
\end{ex}
\end{frame}

\begin{frame}
\frametitle{没有最大素数}
\framesubtitle{证明使用了 \ textit{归谬法}}

\begin{thm}
没有最大素数。
\end{thm}
\begin{proof}[证明]
\begin{enumerate}
\item<1-> 假设~$p$~ 是最大素数.
\item<2-> 令~$q$~ 是前~$p$~ 个数的乘积.
\item<3-> 则~$q$~ 不被它们中的任何一个除尽.
\item<4-> 因此~$q$~ 也是素数而且比~$p$~ 大.\qedhere
\end{enumerate}
\end{proof}
\uncover<4->{证明使用了 \ textit{归谬法}}
\end{frame}

\section{未解决的问题}
```

```
\begin{frame}
  \frametitle{还能做什么?}
  \begin{columns}[t]
    \column{.5\textwidth}
    \begin{block}{已解决的问题}
      有多少素数?
    \end{block}
    \column{.5\textwidth}
    \begin{block}{未解决的问题}
      是否每个偶数都是两个素数之和?\cite{Goldbach1742}
    \end{block}
  \end{columns}
\end{frame}
```

```
\begin{frame}
  \begin{thebibliography}{10}
    \bibitem{Goldbach1742}[Goldback, 1742]
      Christian Goldback.
      \newblock A problem we should try to solve before the ISPN '43 deadline,
      \newblock \emph{Letter to Leonard Euler}, 1742.
    \end{thebibliography}
\end{frame}
```

```
\begin{frame}[fragile]
  \frametitle{寻找素数的 算法 }
```

```
\visible<4->{\texttt{Note the use of \alert{\texttt{std::}}}.}
\end{frame}
```

```
\end{CJK*}
\end{document}
```

```
+-----+
```

```
| 例    子 2|
```

```
+-----+
```

```
% 13-4-2.tex 用 pdfLaTeX 处理
```

```
\documentclass[table]{beamer}
```

```
\mode<presentation>
```

```
{
```

```
  \usetheme{default}
```

```
  % 可供选择的主题参见 beameruserguide.pdf, 第 134 页起
```

```
  % 无导航条的主题: Bergen, Boadilla, Madrid, Pittsburgh, Rochester;
```

```
  % 有树形导航条的主题: Antibes, JuanLesPins, Montpellier;
```

```
  % 有目录竖条的主题: Berkeley, PaloAlto, Goettingen, Marburg, Hannover;
```

```
  % 有圆点导航条的主题: Berlin, Dresden, Darmstadt, Frankfurt, Singapore,
  Szeged;
```

```
  % 有节与小节导航条的主题: Copenhagen, Luebeck, Malmos, Warsaw
```

```
\setbeamercovered{transparent}  
% 如果取消上一行的注解 %, 就会使得被覆盖部分变得透明 (依稀可见)  
}
```

```
\usepackage{times}  
\usepackage{pifont}
```

```
\begin{document}  
\begin{frame}  
\begin{itemize}  
\item<1-| alert@1 > First point.  
\item<2-| alert@2 > Second point.  
\item<3-| alert@3 > Third point.  
\end{itemize}  
\end{frame}
```

```
\newenvironment{ballotenv}
```

```
{\only{%
```

```
\setbeamertemplate{itemize item}{\ding{56}}}%
```

```
\setbeamertemplate{itemize subitem}{\small\ding{56}}%
```

```
\setbeamertemplate{itemize subsubitem}{\footnotesize\ding{56}}}
```

```
}
```

```
\setbeamertemplate{itemize item}{\ding{52}}
```

```
\setbeamertemplate{itemize subitem}{\small\ding{52}}
```

```
\setbeamertemplate{itemize subsubitem}{\footnotesize\ding{52}}
```

```
\begin{frame}
```

```
\begin{itemize}
```

```
\item<1-|
```

ballot@1

> First point.

```
\item<2-|
```

ballot@2

> Second point.

```
\item<3-|
```

ballot@3

> Third point.

`\end{itemize}`

`\end{frame}`

`\end{document}`

+-----+

|TeX 使用指南 |

+-----+

如何把多个文件组装在一起？

所用命令： `\include`, `\includeonly`

TeX Guru 的回答

我想使目录表中显示到 `\subsubsection` 的标题该怎么办？

我想使目录表中显示附录，但前面没有第?? 章字样该怎么办？

所用命令： `\setcounter`, `\tableofcontents` `\addcontentsline`

TeX Guru 的回答

当开始新的一节时，我希望公式自动从 1 开始编号，那该怎么办？

所用命令：\@addtoreset, \makeatletter, \makeatother,

TeX Guru 的回答

章节标题太长该怎么办？

所用命令：\section, \\\

TeX Guru 的回答

在 LaTeX 中，插图单独显示在一页上，而且有相当多的空白。可我如果改成

\begin{figure}[t]，就会把插图显示在文档尾部。这是为什么呢？

所用命令：\renewcommand, \floatpagefraction, \topfraction, \bottomfraction,

\textfraction.

TeX Guru 的回答

在 LaTeX 中，如何使参考文献的编号只是没有方括号的阿拉伯数字或者上标？

所用命令：\renewcommand, \@biblabel, \makeatletter ... \makeatother

TeX Guru 的回答

我用 \pagestyle{empty} 无法去掉所有页的页码，到底怎样可以去掉所有页的页码呢？

所用命令：\pagestyle, \thispagestyle

TeX Guru 的回答

生成了有中文字符的 DVI 文件，该如何把它转化成 PostScript 文件呢？

所用程序：patchdvi, cdvia, dvidrv, dvips32

TeX Guru 的回答

在文件中，想把用其它方法生成的 PostScript 图形包含进来， 可以采用什么方法？

所用命令：\includegraphics, \epsfig

TeX Guru 的回答

CCT EMTex 中的汉字字体只有可怜的 5 种，而市面上却有相当多的 适用于 Windows 中的

True Type 字体，有可能在 CCT 中用这些字体吗？

所用程序：cdvia -c

TeX Guru 的回答

在调试 LaTeX 文件时，如果有一长段中出现了令人讨厌的错误，想把它去掉，但又希望继续

续编译它后面的部分。如果 LaTeX 有块注释功能就太好了，

所用命令：\iffalse, \fi(TeX 命令)

TeX Guru 的回答

在 Book 类中的 `\pagestyle{}` 命令可以生成较为复杂的页眉，可我希望在页眉与正文之间有一横线，该怎么办？

所用命令：`\leftmark`, `\rightmark`, `\@evenhead`, `\@oddhead`, `\makeatletter`,

`\makeatother`

TeX Guru 的回答

+-----+

|TeX 常见问题 1|

+-----+

整理：happytex

1. `\makeatletter` 和 `\makeatother` 的用法？

答：如果需要借助于内部有 `\@` 字符的命令，如 `\@addtoreset`，就需要借助于另两个命令

`\makeatletter`, `\makeatother`。

下面给出使用范例，用它可以实现公式编号与节号的关联。

```
\begin{verbatim}
```

```
\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

16. 辅助的 Package 放到哪个目录下?

答: 不过非 miktex 自带的包最好放在 localtexmf 下, 这样以后升级的时候方便些。

17. 如何用 fancyhdr 宏包实现在页眉下画一粗一细两条横线?

答: 试试下面的命令: \usepackage{fancyhdr}

```
\newcommand{\makeheadrule}{%
```

```
\makebox[0pt][l]{\rule[.7\baselineskip]{\headwidth}{0.8pt}}%
```

```
\rule[.6\baselineskip]{\headwidth}{0.4pt}\vskip-.8\baselineskip}
```

```
\makeatletter \renewcommand{\headrule}{%
```

```
{\if@fancyplain\let\headrulewidth\plainheadrulewidth\fi
```

```
\makeheadrule}}\makeatother
```

18. 能不能改变 Chapter 起始页码仅从奇数页开始?

答: 用 `\documentclass[openany]{book}` 让 `\chapter` 可以从任何页面开始。

19. 如何在文中实现带圈的数字?

答: 1) 使用 `\textcircled{}` 命令, 后面的参数可以是一个字符或者汉字 (配合 CJK), 甚

至也可以是公式, 但是它只会给第一个字符或者汉字加上 > 合适的圆圈, 字体放缩命令

对它也有效, 因此可以利用这点实现两个字符加圈的效果, 但是需要仔细的调整尺寸, 参

考如下命令: `> \Large{\textcircled{\small{12}}}`、`\textcircled{九}`。

2) 使用 `pifont` 宏包, 那里面有很漂亮的带圈的各种数字符号 (仅限于阿拉伯数字, 如果

想使用加圈的中文 “一” 至 “九” 就不行了)。在 `.tex` 文件中尝试下列命令:

$\backslash\text{ding}\{172\}\sim\backslash\text{ding}\{211\}$ 。

3) 在 GB 和 GBK 字库中本来就有 10 个加圈数字形式，可以直接在 CJK 中使用。如果不超过 11

的话应该够用了呵呵：①②③④⑤⑥⑦⑧⑨⑩，这些汉 > 字可以直接使用数字软键盘输

入（以智能 ABC 输入法为例，其实大部分输入法都是一样的）：打开软键盘，选择“数字

序号”即可看到。

20. 如何产生可变长的 Longrightarrow?

答：用 $\backslash\text{xleftarrow}$ and $\backslash\text{xrightarrow}$ ，例如

21. 如何在公式中表示上下堆积关系？

答：在 latex 有一个命令 $\backslash\text{stackre}\{\text{上部符号}\}\{\text{下部符号}\}$ ，但更好的一个是用宏包

amslatex 中命令 $\backslash\text{overset}$ 和 $\backslash\text{underset}$ 。一个例子是

+-----+

|TeX 常见问题 2|

+-----+

1. 怎样在多栏环境中使用单栏图表？

答：如果你在多栏环境中使用， 图形是不会出现的

。如果你用另外的命令， 图形就会超出右边或有

页的上边。解决的方法是把下面命令放在

10。怎样把一个大表格横向放置？

答：用宏包 `\usepackage{rotating}`，然后变换

为

注意这个旋转很可能在你的预览中不会再现。

11。关于 itemize。

答： 清单至多有四级

`\bullet` – `\labelitemi`

`\textbf{--}` – `\labelitemii`

`\ast` – `\labelitemiii`

`\cdot` – `\labelitemiv`

你可重新定义

`\def\labelitemi{\spadesuit}`

`\renewcommand{\labelitemi}{\spadesuit}`

也可在 `itemize` 环境中 `\item` 命令中同时拥有不同的标签

数字序列至多四级顺序可显示，如

1,2 ... – `\enumi`

(a), (b) ... – `\enumii`

i,ii,... – `\enumiii`

A., B. ... – `\enumiv`

你也可指明选项，如

12. 我怎样重新定义 items 之间的距离？

答：用 `miktex2.1` 中的 `mdwlist` 宏包，也可从 CTAN 的

`macros/latex/contrib/supported/mdwtools.zip` 上下载后。

13. 怎样改变行距？

答：用 `double space` 宏包，参数 "coef" 是 `\baselinestretch` 文本中定义数值，则可用命令

14. 怎么定义一个定理型环境？

答：用 theorem 宏包，例子：

```
\newtheorem{howto}{Questions}[section]
```

```
\theoremstyle{change}
```

```
\theorembodyfont{\itshape}
```

```
\theoremheaderfont{\scshape}
```

15. 怎样在 latex 中包括一每列的源文件？

答：例如

```
\floatstyle{boxed}
```

```
\restylefloat{figure}
```

16. 怎样把附录 A 中的公式序号变为 A.1, A.2, B.1, B.2, ... 的形式？

答： \appendix

```
\makeatletter
```

```
\renewcommand\theequation{A.\@arabic\c@equation}
```

```
\makeatother
```

```
\setcounter{equation}{0}
```

或者 `\renewcommand{\theequation}{A.\arabic{equation}}`

17. 怎样把一个 `\verbatim` 放入 `verbatim` 环境中?

答: 用 `fancyvrb` 宏包定义自己的 `verbatim` 环境。

18. 怎样改变主文本中的字体?

答: 命令: `\textnorml`; 申明: `\normalfont\ttfamily`

19. 怎样定义一个命令或申明?

答: 例如一个命令: `\newcommand{\name}{1}{\textsc{#1}}`

一个申明: `\newcommand{\name}{1}{\normalfont\scshape #1}`

20. 怎样合并字体命令?

答: `\textrm{...} {\rmfamily...}` roman familiy

`\textsf{...} {\sffamily...}` sans serif family

`\texttt{...} {\ttfamily...}` typewriter family

`\textmd{...} {\mdseries...}` medium series

`\textbf{...} {\bfseries...}` bold series

`\textup{...} {\upshape...}` upright shape

`\textit{...}` `{\itshape...}` italic shape

`\textsl{...}` `{\slshape...}` slanted shape

`\textsc{...}` `{\scshape...}` small caps shape

size `\tiny` `\scriptsize` `\footnotesize` `\small` `\normalsize` `\large`

`\Large` `\LARGE` `\huge` `\Huge`

`\emph{...}` `{\em ...}` emphasized

`\textnormal{...}` `{\normalfont...}` document font

21. 怎样在页眉和页脚中划一道横线?

答: 用 `fancyhdr` 宏包。

22. 怎样改变章节字体 (section fonts) ?

答: 用 `titlesec` 宏包, 例如

`\usepackage[nops,sf,small,compact]{titlesec}`

23. 怎样做一个页面设置?

答: `\usepackage{fancyhdr}`

`\pagestyle{fancy}`

```
\fancyhead{} %clear all fields
```

```
\fancyhead[RO, LE]{\bfseries \LaTeXe{} HOWTO}
```

```
\fancyfoot[LE, RO]{\thepage\ of \pageref{LastPage}}
```

```
\fancyfoot[LO, CE]{}%
```

```
\fancyfoot[CO, RE]{Techical Report}
```

```
\renewcommand{\headrulewidth}{0.4pt}
```

```
\renewcommand{\footrulewidth}{0.4pt}
```

24. 怎样实现 "page M of N" 的形式?

答：用 lastpage 宏包，任何时候你都可在文本中用 \pageref{LastPage}，只是须注意当用了 fancyhdr 宏包时，lastpage 宏包就须在它后面。

25. 怎样实现下划线的标题?

答：

```
\def\@evenhead{\makebox[0pt][l]{\rule[-0.15cm]{\textwidth}{0.1mm}}\hbox{}%
```

```
\textbf{\thepage\hfil\leftmark}}%
```

```
\def\@oddhead{\makebox[0pt][l]{\rule[-0.15cm]{\textwidth}{0.1mm}}\hbox{}%
```

`{\rightmark}\hfil\textbf{\thepage}%`

26。怎样把一个浮动对象固定下来？

答：例如

30。把一幅 .eps 横向放置后，怎样使标题变为相应的形式？

答： `\usepackage{rotating}`

和

+-----+

|TeX 常见问题 3|

+-----+

压缩空间

1。页面设置

a4 会给你一个较小的页面，为了使用更多的控制，可用 geometry 宏包和命令 `\layout`

。

2。改变长度

在 latex 里改变长度的命令是 `\addtolength` 和 `\setlength`。latex 可认的是 cm, mm,

in 和 pt. 变量可设为负数。一个例子是 `\addtolength{\parindent}{-5mm}`

一些有用的变量是：

页面设置

`\columnsep`: 列间距

`\topmargin`: 页眉到页边的距离

`\topskip`: 页眉与正文的距离

`\textheight`: 正文的高度

`\textwidth`: 文本的宽度

`\oddsidemargin`: 奇数页的左面页边距

`\evensidemargin` : 偶数页的左面页边距

段落

`\parindent`: 段落缩进距离

`\parskip`: 段落间的距离

浮动图表

`\floatsep`: 浮动对象之间的距离

`\textfloatsep`: 最后一个浮动对象顶端或第一个浮动对象底端与正文之间的距离

`\intextsep` : 文中浮动顶端与底端所留的距离

`\dbltextfloatsep` 是在双列输出时用 `\textfloatsep` 的数值

`\dblfloatsep` 是在双列输出时用 `\floatsep` 的数值

`\abovecaptionskip`: 标题上方的距离

`\belowcaptionskip`: 标题下方的距离

数学公式

`\abovedisplayskip`: 公式前的距离

`\belowdisplayskip`: 公式后面的距离

`\arraycolsep`: 在一个 `array` 中列之间的空白长度

列表

`\topsep`: 第一个 item 和前面版落间的距离

`\partopsep`: 当在一个新页开始时加到 `\topsep` 的额外空间

`\itemsep`: 连续 items 之间的距离。

3. 环境

宏包 `atbeginend.sty` 提供了 `\BeforeBegin{environment}{code-to-execute}`,

`\AfterEnd` 等等命令。这些命令可去掉环境外面和内部的空间。例如

`\AfterBegin{itemize}{\addtolength{\itemsep}{-\baselineskip}}` 就在环境内以重新

设置 `\itemsep` 来压缩 items 之间的距离。

`mdwlist` 宏包有一个 `itemize*` 环境; `paralist` 宏包提供了一个 item 之间距离压缩环境

和一个使每个 item 不在新的一段开始的 `inparaenum` 环境。

4. 图表

为了节约空间, 可用 `subfigure` 宏包把多幅图形放在一人图形环境中, 也可用

`floatflt` 宏包把图形用文本包围起来。

当有图表时, LaTeX 缺省值不会在超过 70% 的页中加文本。这个可有

问题: LaTeX 中如何使公式编号具有 (1.2) 这种格式?

回答: It is default style in book. In article style,

use the following command

```
\documentclass{article}
```

```
\usepackage{amsmath}
```


`\numberwithin{equation}{section}`

问题: tex 中花体字母 N 怎么输入?

回答: 不是花体字母 N, 是标准的 N . 要用花体的话, \mathcal{N}

问题: 在 LaTeX 中怎样生成双线的方括号? 哪儿有 stmaryrd

回答:

<ftp://159.226.47.19/pub/Tex/Package/latex/contrib/other/misc/>

问题: 请问如何把 word 文档转换成 latex 的格式?

回答: 用 mathtype4.0, 双击公式可打开公式编辑器,

设定菜单 Preferences->Translators

Translate to another language:

选 latex2.09 and later

uncheck 'include translator name' and 'include mathtype data'

选中整个公式, ^c, 到 notepad 里面 ^V 就行了

我现在就这么干的。不过熟悉之后, 直接敲 tex 公式也不慢。

from: <http://zh.wikipedia.org/wiki/Help:MATH>

<http://blog.csdn.net/anxiaoxi45/article/details/39449445>

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎^{beta}，[点击查看详细说明](#)

