

LaTeX 数学符号与语法快速参考

基于 LaTeX Cheatsheet

更新：2026 年 2 月 2 日

目录

前言	6
1 入门	6
1.1 数学模式	6
1.2 公式显示效果对比	7

2	希腊字母	9
2.1	小写希腊字母	9
2.2	变体形式	9
2.3	大写希腊字母	10
3	上下标与修饰符号	11
3.1	上标与下标	11
3.2	重音与修饰符号	11
4	分数与根号	13
4.1	分数的多种写法	13
4.2	根号	14
5	运算符	15
5.1	二元运算符	15
5.2	大型运算符	15

6	关系符号	17
6.1	等式与不等式	17
6.2	集合论符号	17
6.3	逻辑符号	18
6.4	常用数集	19
7	括号与分隔符	20
7.1	基本括号类型	20
7.2	自动调整大小	20
8	矩阵与数组	22
8.1	矩阵环境对比	22
8.2	省略号用法	22
9	常用数学环境	25
9.1	equation 环境	25

9.2 align 环境	25
9.3 cases 环境	26
10 注解与标注	27
11 常用数学函数	28
12 高级数学主题	29
12.1 泛函分析	29
12.2 神经网络与机器学习	30
13 TikZ 绘图	31
13.1 基本图形	31
13.2 节点与连线	31
13.3 神经网络可视化	32
13.4 PGFPlots 函数图	32

14 Font Awesome 图标	33
14.1 常用图标速查	33
14.2 信息提示框	33
15 LaTeX3 (expl3) 编程	35
15.1 命名约定	35
15.2 xparse 文档命令	35
16 参考资源	37

前言

📖 关于本教程

本教程汇集了 \LaTeX 数学排版中最常用的符号、语法和技巧。无论你是刚接触 \LaTeX 的新手，还是需要快速查阅的老用户，这份参考都能帮助你高效地完成数学公式排版。

每个示例都提供「代码」与「效果」的直观对比，让学习更加轻松。

1 入门

\LaTeX 是一种基于 \TeX 的排版系统，广泛用于学术论文、技术文档和书籍的排版。本章将介绍数学公式排版的基础知识，帮助你快速上手。

1.1 数学模式

\LaTeX 中的数学公式有两种基本形式：

- **行内公式**：使用 $\$...\$$ 或 $\backslash(...\backslash)$ 包裹，公式与文字融为一体
- **行间公式**：使用 $$$...$$$ 或 $\backslash[...\backslash]$ 包裹，公式独占一行居中显示

例 1.1 (行内公式) 代码：质能方程 $E=mc^2$ 是物理学的基础公式。

效果：质能方程 $E = mc^2$ 是物理学的基础公式。

例 1.2 (行间公式) 代码：

```
\[ f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2\pi i \xi x} d\xi \]
```

效果：

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2\pi i \xi x} d\xi$$

注 行间公式更适合展示重要公式，而行内公式适合嵌入文字中的简短表达式。推荐使用 `\[...\]` 而非 `$$...$$`，前者是 \LaTeX 标准写法。

1.2 公式显示效果对比

💡 同一个公式在行内和行间模式下显示效果有显著差异，尤其是分数和积分。

例 1.3 (强制切换显示样式) 使用 `\displaystyle` 在行内强制大尺寸显示：

行内大分数： $\frac{a}{b}$

代码： `\displaystyle\frac{a}{b}`

表 1: 行内模式 vs 行间模式显示对比

公式类型	行内模式	行间模式
分数	$\frac{a}{b}$	$\frac{a}{b}$
求和	$\sum_{i=1}^n i$	$\sum_{i=1}^n i$
积分	$\int_0^1 x \, dx$	$\int_0^1 x \, dx$
极限	$\lim_{x \rightarrow 0} f(x)$	$\lim_{x \rightarrow 0} f(x)$

2 希腊字母

希腊字母在数学和物理中无处不在——从 α, β, γ 表示角度，到 Ω 表示电阻，再到 π 这个神奇的圆周率。掌握这些字母的输入方式是使用 \LaTeX 的第一步。

2.1 小写希腊字母

表 2: 小写希腊字母速查表

预览	代码	预览	代码	预览	代码	预览	代码
α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>	δ	<code>\delta</code>
ϵ	<code>\epsilon</code>	ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>
ι	<code>\iota</code>	κ	<code>\kappa</code>	λ	<code>\lambda</code>	μ	<code>\mu</code>
ν	<code>\nu</code>	ξ	<code>\xi</code>	π	<code>\pi</code>	ρ	<code>\rho</code>
σ	<code>\sigma</code>	τ	<code>\tau</code>	υ	<code>\upsilon</code>	ϕ	<code>\phi</code>
χ	<code>\chi</code>	ψ	<code>\psi</code>	ω	<code>\omega</code>		

2.2 变体形式

部分希腊字母有变体形式，在不同场景下使用：

表 3: 希腊字母变体对比

标准形式	代码	变体形式	代码	应用场景
ϵ	<code>\epsilon</code>	ε	<code>\varepsilon</code>	分析学常用 ε
θ	<code>\theta</code>	ϑ	<code>\vartheta</code>	手写体风格
ϕ	<code>\phi</code>	φ	<code>\varphi</code>	φ 表示黄金比例
ρ	<code>\rho</code>	ϱ	<code>\varrho</code>	带卷曲尾巴
σ	<code>\sigma</code>	ς	<code>\varsigma</code>	希腊词尾形式

2.3 大写希腊字母

表 4: 大写希腊字母（仅显示与拉丁字母不同的）

预览	代码	预览	代码	预览	代码	预览	代码
Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>	Λ	<code>\Lambda</code>
Ξ	<code>\Xi</code>	Π	<code>\Pi</code>	Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>
Φ	<code>\Phi</code>	Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>		

注 大写希腊字母中 A, B, E, H, I, K, M, N, O, P, T, X, Z 与拉丁字母相同，因此直接输入即可，无需特殊命令。

3 上下标与修饰符号

掌握了希腊字母后，下一步是学习如何给符号添加上下标和各种修饰——从简单的幂指数到复杂的向量表示，这些语法是构建公式的基础砖块。

3.1 上标与下标

表 5: 上下标语法

效果	代码	效果	代码
a^2	<code>a^2</code>	a_i	<code>a_i</code>
a^{n+1}	<code>a^{n+1}</code>	$a_{i,j}$	<code>a_{i,j}</code>
x_n^2	<code>x_n^2</code>	${}_1^2X_3^4$	<code>{_1^2X_3^4}</code>

⚠ 多字符上下标必须用花括号包裹！`a^10` 显示为 a^10 （只有 1 是上标），而 `a^{10}` 才正确显示为 a^{10} 。

3.2 重音与修饰符号

数学中常用重音符号表示特殊含义，如导数、平均值、向量等：

表 6: 常用重音符号

效果	代码	用途	效果	代码	用途
\hat{a}	<code>\hat{a}</code>	估计量	\widehat{AB}	<code>\widehat{AB}</code>	宽帽子
\bar{x}	<code>\bar{x}</code>	平均值	\overline{xyz}	<code>\overline{xyz}</code>	共轭
\vec{v}	<code>\vec{v}</code>	向量	\overrightarrow{AB}	<code>\overrightarrow{AB}</code>	有向线段
\dot{x}	<code>\dot{x}</code>	一阶导	\ddot{x}	<code>\ddot{x}</code>	二阶导
\tilde{a}	<code>\tilde{a}</code>	近似	\widetilde{abc}	<code>\widetilde{abc}</code>	宽波浪

例 3.1 (物理中的典型应用)

牛顿第二定律: $\vec{F} = m\ddot{\vec{r}}$

样本均值: $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$

傅里叶变换: $\hat{f}(\xi) = \int f(x) e^{-2\pi i x \xi} dx$

4 分数与根号

分数和根号是数学表达式中最常见的结构。L^AT_EX 提供了多种方式来精确控制它们的显示效果。

4.1 分数的多种写法

表 7: 分数表示方法对比

效果	代码	说明
$\frac{a}{b}$	<code>\frac{a}{b}</code>	标准分数，大小自适应
$\frac{a}{b}$	<code>\dfrac{a}{b}</code>	强制大尺寸
$\frac{a}{b}$	<code>\tfrac{a}{b}</code>	强制小尺寸
a/b	<code>a/b</code>	斜线分数
$\frac{a}{1 + \frac{1}{b}}$	<code>\cfrac{...}</code>	连分数专用

例 4.1 (分数在不同场景的选择) 行内使用 `\tfrac` 更紧凑：概率 $\frac{1}{6}$ 表示骰子某面朝上

贝叶斯公式 (行间用 `\dfrac`):

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

黄金比例连分数:

$$\phi = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{\ddots}}}$$

4.2 根号

表 8: 根号语法

效果	代码	效果	代码
\sqrt{x}	<code>\sqrt{x}</code>	$\sqrt[3]{x}$	<code>\sqrt[3]{x}</code>
$\sqrt[n]{x}$	<code>\sqrt[n]{x}</code>	$\sqrt{x^2 + y^2}$	<code>\sqrt{x^2+y^2}</code>

例 4.2 (求根公式) 二次方程 $ax^2 + bx + c = 0$ 的求根公式:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

5 运算符

从基本的加减乘除到复杂的积分求和，运算符是公式的核心。本节将系统介绍各类运算符及其正确用法。

5.1 二元运算符

表 9: 二元运算符

符号	代码	符号	代码	符号	代码	符号	代码
+	<code>+</code>	−	<code>-</code>	×	<code>\times</code>	÷	<code>\div</code>
±	<code>\pm</code>	∓	<code>\mp</code>	·	<code>\cdot</code>	*	<code>\ast</code>
⊕	<code>\oplus</code>	⊗	<code>\otimes</code>	⊙	<code>\odot</code>	○	<code>\circ</code>

5.2 大型运算符

大型运算符在行内和行间模式下会自动调整上下标位置：

表 10: 大型运算符对比

运算符	行内模式	行间模式
求和 \sum	$\sum_{i=1}^n a_i$	$\sum_{i=1}^n a_i$
积分 \int	$\int_a^b f(x) dx$	$\int_a^b f(x) dx$
乘积 \prod	$\prod_{i=1}^n i$	$\prod_{i=1}^n i$
极限 \lim	$\lim_{x \rightarrow \infty} f(x)$	$\lim_{x \rightarrow \infty} f(x)$

例 5.1 (多重积分与曲面积分)

二重积分: $\iint_D f(x, y) dA$

三重积分: $\iiint_V \rho dV$

曲线积分: $\oint_C \vec{F} \cdot d\vec{r}$

6 关系符号

关系符号用于表示数学对象之间的比较、等价和包含关系。正确使用这些符号可以使数学表达更加精确规范。

6.1 等式与不等式

表 11: 基本关系符号

符号	代码	符号	代码	符号	代码	符号	代码
$=$	<code>=</code>	\neq	<code>\neq</code>	\equiv	<code>\equiv</code>	\approx	<code>\approx</code>
$<$	<code><</code>	$>$	<code>></code>	\leq	<code>\leq</code>	\geq	<code>\geq</code>
\ll	<code>\ll</code>	\gg	<code>\gg</code>	\sim	<code>\sim</code>	\propto	<code>\propto</code>

6.2 集合论符号

集合论是现代数学的基础语言：

表 12: 集合关系与运算

符号	代码	符号	代码	符号	代码	符号	代码
\in	<code>\in</code>	\notin	<code>\notin</code>	\cup	<code>\cup</code>	\cap	<code>\cap</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\bigcup	<code>\bigcup</code>	\bigcap	<code>\bigcap</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\setminus	<code>\setminus</code>	\emptyset	<code>\emptyset</code>

例 6.1 (集合论典型表达式)

$$A \cup B = \{x : x \in A \vee x \in B\} \quad (\text{并集})$$

$$A \cap B = \{x : x \in A \wedge x \in B\} \quad (\text{交集})$$

$$A \setminus B = \{x : x \in A \wedge x \notin B\} \quad (\text{差集})$$

6.3 逻辑符号

表 13: 逻辑运算符与量词

符号	代码	符号	代码	符号	代码	符号	代码
\forall	<code>\forall</code>	\exists	<code>\exists</code>	\nexists	<code>\nexists</code>	\neg	<code>\neg</code>
\wedge	<code>\wedge</code>	\vee	<code>\vee</code>	\implies	<code>\implies</code>	\iff	<code>\iff</code>

6.4 常用数集

表 14: 常用数集（需要 `amssymb` 宏包）

符号	代码	含义
\mathbb{N}	<code>\mathbb{N}</code>	自然数 $\{0, 1, 2, \dots\}$
\mathbb{Z}	<code>\mathbb{Z}</code>	整数
\mathbb{Q}	<code>\mathbb{Q}</code>	有理数
\mathbb{R}	<code>\mathbb{R}</code>	实数
\mathbb{C}	<code>\mathbb{C}</code>	复数

7 括号与分隔符

合适的括号大小对于公式的可读性至关重要。 \LaTeX 提供了自动和手动两种方式来控制括号大小。

7.1 基本括号类型

表 15: 各种括号与分隔符

效果	代码	效果	代码	效果	代码
$()$	<code>(~)</code>	$[]$	<code>[~]</code>	$\{\}$	<code>\{~\}</code>
$\langle\rangle$	<code>\langle~\rangle</code>	$ $	<code> ~ </code>	$ $	<code>\ ~\ </code>
$\lceil\rceil$	<code>\lceil~\rceil</code>	$\lfloor\rfloor$	<code>\lfloor~\rfloor</code>		

7.2 自动调整大小

使用 `\left` 和 `\right` 可以让括号自动匹配内容高度：

例 7.1 (自动 vs 手动大小调整) 不调整 vs 自动调整：

$$\left(\frac{1}{2}\right) \quad \text{vs} \quad \left(\frac{1}{2}\right)$$

手动指定大小（从小到大）：

$$((((x))))$$

💡 `\left` 和 `\right` 必须成对。如需单边括号，用 `\left.` 或 `\right.` 作为隐形占位符。

例 7.2 (单边括号示例（求导数在某点的值）)

$$\left.\frac{df}{dx}\right|_{x=0}$$

代码： `\left. \frac{d f}{d x} \right|_{x=0}`

8 矩阵与数组

矩阵是线性代数的核心对象。L^AT_EX 的 `amsmath` 宏包提供了多种矩阵环境。

8.1 矩阵环境对比

例 8.1 (矩阵输入语法)

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

8.2 省略号用法

表 16: 矩阵环境一览

环境名	效果	括号类型
<code>matrix</code>	$\begin{matrix} a & b \\ c & d \end{matrix}$	无括号
<code>pmatrix</code>	$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	圆括号
<code>bmatrix</code>	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	方括号
<code>Bmatrix</code>	$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$	花括号
<code>vmatrix</code>	$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	单竖线 (行列式)
<code>Vmatrix</code>	$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$	双竖线 (范数)

表 17: 省略号类型

效果	代码	用途
...	<code>\cdots</code>	水平居中省略
...	<code>\ldots</code>	水平底部省略
⋮	<code>\vdots</code>	垂直省略
⋯	<code>\ddots</code>	对角线省略

9 常用数学环境

单行公式用 `equation`，多行对齐用 `align`，分段函数用 `cases`——选择合适的环境可以让公式排版更加专业。

9.1 `equation` 环境

带编号的单行公式：

```
\begin{equation}
E = mc^2
\end{equation}
```

效果：

$$E = mc^2 \quad (1)$$

9.2 `align` 环境

多行对齐公式，使用 `&` 标记对齐点：

```
\begin{align}
(x+y)^2 &= (x+y)(x+y) \\
&= x^2 + 2xy + y^2
\end{align}
```

```
\end{align}
```

效果：

$$(x + y)^2 = (x + y)(x + y) \quad (2)$$

$$= x^2 + 2xy + y^2 \quad (3)$$

9.3 cases 环境

分段函数定义：

```
\[ |x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}
```

效果：

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

10 注解与标注

有时需要在公式中添加删除线、方框或括号标注来强调特定部分。

表 18: 注解与标注符号

效果	代码	说明
$\cancel{5}$	<code>\cancel{5}</code>	斜线删除
$\bcancel{5}$	<code>\bcancel{5}</code>	反斜线删除
\xcancel{ABC}	<code>\xcancel{ABC}</code>	交叉删除
\sout{abc}	<code>\sout{abc}</code>	水平删除线
$\boxed{\pi = \frac{c}{d}}$	<code>\boxed{...}</code>	方框标注
$\overbrace{a+b}^{\text{sum}}$	<code>\overbrace{...}^{\{...\}}</code>	上括号标注
$\underbrace{a+b}_{\text{sum}}$	<code>\underbrace{...}_{\{...\}}</code>	下括号标注

例 10.1 (公式推导中的标注)

$$\overbrace{a+b+c}^{\text{后算}} = a + (b+c)$$
$$\underbrace{\frac{1}{x}}_{\text{先算}} \cdot x = 1$$

11 常用数学函数

L^AT_EX 预定义了许多数学函数名，使用这些命令可以确保函数名以正体显示。

表 19: 预定义数学函数

函数	代码	函数	代码	函数	代码	函数	代码
sin	<code>\sin</code>	cos	<code>\cos</code>	tan	<code>\tan</code>	cot	<code>\cot</code>
arcsin	<code>\arcsin</code>	arccos	<code>\arccos</code>	arctan	<code>\arctan</code>		
log	<code>\log</code>	ln	<code>\ln</code>	exp	<code>\exp</code>	det	<code>\det</code>
gcd	<code>\gcd</code>	min	<code>\min</code>	max	<code>\max</code>	lim	<code>\lim</code>

例 11.1 (函数正体 vs 斜体对比) 正确: $\sin(x), \log_2(n)$

错误: $\sin(x), \log_2(n)$

注意: 斜体 \sin 看起来像三个变量 $s \cdot i \cdot n$ 相乘

12 高级数学主题

本节介绍在专业数学文档中常见的符号，涵盖泛函分析和神经网络等领域。

12.1 泛函分析

表 20: 范数与内积

效果	代码	说明
$\ x\ $	<code>\ x\ </code>	范数
$\ x\ _p$	<code>\ x\ _p</code>	L^p 范数
$\langle x, y \rangle$	<code>\langle x, y \rangle</code>	内积
$x \perp y$	<code>x \perp y</code>	正交

例 12.1 (泛函分析典型公式)

$$\|f\|_{L^p} = \left(\int_{\Omega} |f(x)|^p dx \right)^{1/p}$$
$$\langle f, g \rangle_{L^2} = \int_{\Omega} f(x) \overline{g(x)} dx$$

表 21: 神经网络常用符号

效果	代码	说明
\mathbf{W}	<code>\mathbf{W}</code>	权重矩阵
$\hat{\mathbf{y}}$	<code>\hat{\mathbf{y}}</code>	预测值
\mathcal{L}	<code>\mathcal{L}</code>	损失函数
$\nabla_{\theta}\mathcal{L}$	<code>\nabla_{\theta} \mathcal{L}</code>	梯度
\odot	<code>\odot</code>	Hadamard 积

12.2 神经网络与机器学习

例 12.2 (神经网络核心公式)

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (\text{Sigmoid})$$

$$\text{ReLU}(x) = \max(0, x)$$

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

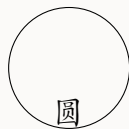
$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} \mathcal{L} \quad (\text{SGD})$$

13 TikZ 绘图

TikZ 是 \LaTeX 中最强大的绘图包，可以创建各种矢量图形。

13.1 基本图形

例 13.1 (基本形状绘制)



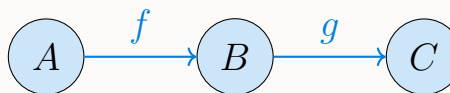
矩形



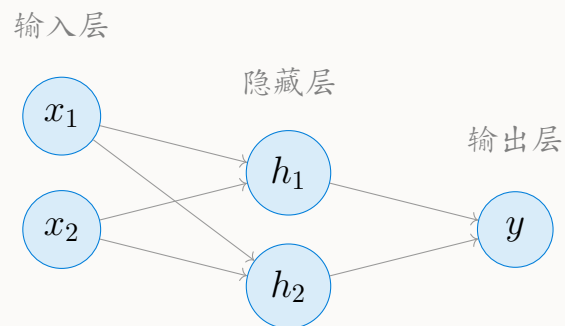
三角形

13.2 节点与连线

例 13.2 (流程图节点)

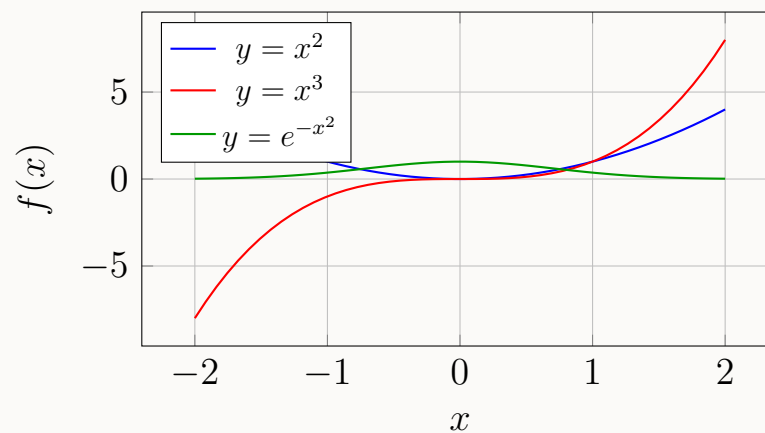


13.3 神经网络可视化



例 13.3 (简单神经网络架构)

13.4 PGFPlots 函数图















例 13.4 (函数图)

14 Font Awesome 图标

Font Awesome 5 提供了上千个矢量图标，可以让你的文档更加生动专业。

14.1 常用图标速查

表 22: 常用图标

图标	代码	图标	代码	图标	代码	图标	代码
	<code>\faHome</code>		<code>\faUser</code>		<code>\faEnvelope</code>		<code>\faPhone</code>
	<code>\faSearch</code>		<code>\faCog</code>		<code>\faCheck</code>		<code>\faTimes</code>
	<code>\faGithub</code>		<code>\faPython</code>		<code>\faRust</code>		<code>\faDocker</code>

14.2 信息提示框

利用图标可以创建信息提示效果：

例 14.1 (图标与文字组合)		信息提示	<code>\faInfoCircle</code>
		小技巧	<code>\faLightbulb</code>
		警告	<code>\faExclamationTriangle</code>
		成功	<code>\faCheckCircle</code>
		错误	<code>\faTimesCircle</code>

15 LaTeX3 (expl3) 编程

LaTeX3 是 L^AT_EX 的现代化编程层，提供了一致的命名约定和强大的编程能力。

15.1 命名约定

expl3 使用 模块_函数:参数说明 的命名格式：

表 23: expl3 模块简介

模块	描述	示例
int	整数	<code>\int_step_inline:nn</code>
tl	Token List	<code>\tl_set:Nn</code>
str	字符串	<code>\str_if_eq:nnTF</code>
seq	序列	<code>\seq_push:Nn</code>
fp	浮点数	<code>\fp_eval:n</code>

15.2 xparse 文档命令

xparse 提供了灵活的命令定义方式：

表 24: xparse 参数类型

类型	描述	示例用法
m	必选参数	{内容}
o	可选参数	[选项]
0{默认}	带默认值可选参数	省略时用默认值
s	星号标记	*

16 参考资料

🌐 L^AT_EX 官方网站: <https://www.latex-project.org/>

📖 Overleaf 学习资源: <https://www.overleaf.com/learn>

🔗 KaTeX 支持符号: <https://katex.org/docs/supported.html>

🎨 TikZ 示例库: <https://tikz.dev/>

🔗 Font Awesome 图标: <https://fontawesome.com/>

🔗 ElegantNote 模板: <https://github.com/ElegantLaTeX/ElegantNote>

♥ Happy T_EXing!