

分类号 \_\_\_\_\_

编 号 \_\_\_\_\_

UDC \_\_\_\_\_

密 级 \_\_\_\_\_



南方科技大学

# 本科生毕业设计（论文）

题 目： \_\_\_\_\_ 南方科技大学学位论文模板

\_\_\_\_\_ sustech-ug-thesis-typst v0.0.1

姓 名： \_\_\_\_\_ 张三

学 号： \_\_\_\_\_ 12010000

学 院： \_\_\_\_\_ 某个学院

专 业： \_\_\_\_\_ 某个专业

指导教师： \_\_\_\_\_ 李四 教授

二零二四年六月 X 日

# 诚信承诺书

1. 本人郑重承诺所呈交的毕业设计（论文），是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料均真实可靠。
2. 除文中已经注明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究作出重要贡献的个人和集体，均已在文中以明确的方式标明。
3. 本人承诺在毕业论文（设计）选题和研究内容过程中没有抄袭他人研究成果和伪造相关数据等行为。
4. 在毕业论文（设计）中对侵犯任何方面知识产权的行为，由本人承担相应的法律责任。

作者签名：

二零二四年六月 X 日

## 摘要

本文介绍了 sustech-ug-thesis-typst 文档模板所提供的功能。

**关键词：** Typst, 模板

# SUSTech thesis document template

## sustech-ug-thesis-typst v0.0.1

San Zhang (Some Major)

Directed by Prof. Si Li

### Abstract

This document introduces the features of the sustech-ug-thesis-typst template.

**KEYWORDS:** Typst, Template

### 目录

<b>第一章 基本功能</b>	<b>6</b>
1.1 标题 .....	6
1.1.1 三级标题 .....	7
1.2 粗体与斜体 .....	7
1.3 脚注 .....	8
1.4 图片 .....	8
1.5 表格 .....	9
1.6 公式 .....	11
1.7 代码块 .....	12
1.8 参考文献 .....	14

<b>第二章 理论</b>	<b>15</b>
2.1 理论一 .....	15
2.2 理论二 .....	15
<b>第三章 展望</b>	<b>16</b>
<b>附录 A 关于 Typst</b>	<b>16</b>
A.1 在附录中插入图片和公式等 .....	17
A.2 Typst 的开发者 .....	19
<b>附录 B 单独的章节</b>	<b>19</b>
<b>南方科技大学学位论文原创性声明和使用授权说明</b>	<b>20</b>

## 插图

1.1 Typst 网页版界面 .....	9
A.1 Typst 网页版界面 .....	17

## 表格

1.1 答辩委员会名单 .....	9
1.2 booktab 示例 .....	11

## 代码

1.1 默认表格 .....	9
1.2 一个简单的 Python 程序 .....	13

1.3codeblock 命令的实现 ..... 13

A.1一个简单的 Rust 程序 ..... 18

# 第一章 基本功能

## 1.1 标题

Typst 中的标题使用 `=` 表示，其后跟着标题的内容。`=` 的数量对应于标题的级别。

除了这一简略方式，也可以通过 `heading` 函数自定义标题的更多属性。具体可以参考文档中的有关内容。

下面是一个示例：

代码	渲染结果
<code>#heading(level: 2, numbering: none, outlined: false, "二级标题")</code>	二级标题
<code>#heading(level: 3, numbering: none, outlined: false, "三级标题")</code>	三级标题
<code>#heading(level: 4, numbering: none, outlined: false, "四级标题")</code>	四级标题
<code>#heading(level: 5, numbering: none, outlined: false, "五级标题")</code>	五级标题

需要注意的是, 这里的样式经过了本模板的一些定制, 并非 Typst 的默认样式。

### 1.1.1 三级标题

#### 1.1.1.1 四级标题

本模板目录的默认最大深度为 3, 即只有前三级标题会出现在目录中。如果需要更深的目录, 可以更改 `outlinedepth` 设置。

## 1.2 粗体与斜体

与 Markdown 类似, 在 Typst 中, 使用 `*...*` 表示粗体, 使用 `_..._` 表示斜体。下面是一个示例:

代码	渲染结果
<code>*bold*</code> and <code>_italic_</code> are very simple.	<b>bold</b> and <i>italic</i> are very simple.

由于绝大部分中文字体只有单一字形, 这里遵循 PKUTHSS 的惯例, 使用**黑体**表示粗体, 楷体表示斜体。但需要注意的是, 由于语法解析的问题, `*...*` 和 `_..._` 的前后可能需要空格分隔, 而这有时会导致不必要的空白。如果不希望出现这一空白, 可以直接采用 `#strong` 或 `#emph`。

代码	渲染结果
----	------

对于中文情形， <b>*使用 \* 加粗*</b> 会导致额外的空白， <code>#strong</code> [使用 <code>\#strong</code> 加粗] 则不会。	对于中文情形， <b>使用 * 加粗</b> 会导致额外的空白， <b>使用 #strong 加粗</b> 则不会。
---	--

### 1.3 脚注

从 v0.4 版本开始，Typst 原生支持了脚注功能。本模板中，默认每一章节的脚注编号从 1 开始。

代码	渲染结果
Typst 支持添加脚注 <code>#footnote</code> [这是一个脚注。]。	Typst 支持添加脚注 <sup>1</sup> 。

### 1.4 图片

在 Typst 中插入图片的默认方式是 `image` 函数。如果需要给图片增加标题，或者在文章中引用图片，则需要将其放置在 `figure` 中，就像下面这样：

代码	渲染结果
----	------

---

<sup>1</sup>这是一个脚注。



```
#figure(  
  image("images/1-writing-app.png", width: 100%),  
  caption: "Typst 网页版界面",  
) <web>
```

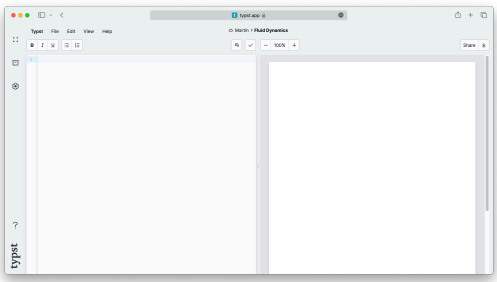


图 1.1 Typst 网页版界面

图 1.1 展示了 Typst 网页版的界面。更多有关内容，可以参考附录 A。节 A.2 中介绍了 Typst 的主要开发者。代码中的 `<web>` 是这一图片的标签，可以在文中通过 `@web` 来引用。

1.5 表格

在 Typst 中，定义表格的默认方式是 `table` 函数。但如果需要给表格增加标题，或者在文章中引用表格，则需要将其放置在 `figure` 中，就像下面这样：

代码	渲染结果								
代码 1.1 默认表格	<div>表 1.1 答辩委员会名单</div> <table><tr><th>姓名</th><th>职称</th><th>工 作 单 位</th><th>职责</th></tr><tr><td>李四</td><td>教授</td><td>北 京 大 学</td><td>主席</td></tr></table>	姓名	职称	工 作 单 位	职责	李四	教授	北 京 大 学	主席
姓名	职称	工 作 单 位	职责						
李四	教授	北 京 大 学	主席						

```
#figure(  
  table(  
    columns: (auto, auto, auto,  
auto),  
    inset: 10pt,  
    align: horizon,  
      [*姓名*],[*职称*],[*工作单位  
*],[*职责*],  
      [李四],[教授],[北京大学],[主  
席],  
      [王五],[教授],[北京大学],[成  
员],  
      [赵六],[教授],[北京大学],[成  
员],  
      [钱七],[教授],[北京大学],[成  
员],  
      [孙八],[教授],[北京大学],[成  
员],  
    ),  
    caption: "答辩委员会名单",  
  ) <table>
```

王五	教授	北 京 大学	成员
赵六	教授	北 京 大学	成员
钱七	教授	北 京 大学	成员
孙八	教授	北 京 大学	成员

对应的渲染结果如表 1.1 所示。代码中的 <table> 是这一表格的标签，可以在文中通过 @table 来引用。

默认的表格不是特别美观，本模板中提供了 booktab 函数用于生成三线表，表 1.2 是一个示例。代码中的 <booktab> 是这一表格的标签，可以在文中通过 @booktab 来引用。

代码	渲染结果												
<pre>#booktab(  width: 100%,  aligns: (left, center, right),  columns: (1fr, 1fr, 1fr),  caption: [``booktab` 示例],  [左对齐], [居中], [右对齐],  [4], [5], [6],  [7], [8], [9],  [10], [], [11],  ) &lt;booktab&gt;</pre>	<div>表 1.2 booktab 示例</div> <table><tr><th>左对齐</th><th>居中</th><th>右对齐</th></tr><tr><td>4</td><td>5</td><td>6</td></tr><tr><td>7</td><td>8</td><td>9</td></tr><tr><td>10</td><td></td><td>11</td></tr></table>	左对齐	居中	右对齐	4	5	6	7	8	9	10		11
左对齐	居中	右对齐											
4	5	6											
7	8	9											
10		11											

# 1.6 公式

式 (1.1)是一个公式。代码中的 `<eq>` 是这一公式的标签，可以在文中通过 `@eq` 来引用。

代码	渲染结果
<code>\$ E = m c^2 \$ &lt;eq&gt;</code>	$E = mc^2$ (1.1)

式 (1.2)是一个多行公式。

代码	渲染结果
<pre>\$ \sum_{k=0}^n k   &amp;= 1 + \dots + n \setminus   &amp;= (n(n+1)) / 2 \$ &lt;eq2&gt;</pre>	$\sum_{k=0}^n k = 1 + \dots + n$ $= \frac{n(n+1)}{2}$ <p>(1.2)</p>

式 (1.3)到式 (1.6)中给出了更多的示例。

代码	渲染结果
<code>\$ frac(a^2, 2) \$ &lt;eq3&gt;</code>	$\frac{a^2}{2} \quad (1.3)$
<code>\$ vec(1, 2, delim: "[") \$</code>	$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (1.4)$
<code>\$ mat(1, 2; 3, 4) \$</code>	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad (1.5)$
<code>\$ lim_x =     op("lim", limits: #true)_x \$ &lt;eq6&gt;</code>	$\lim_x = \lim_x \quad (1.6)$

## 1.7 代码块

像 Markdown 一样，我们可以在文档中插入代码块：

代码	渲染结果
<code>```c  int main() {     printf("Hello, world!");     return 0; } ```</code>	<pre>int main() {     printf("Hello, world!");     return 0; }</pre>

如果想要给代码块加上标题，并在文章中引用代码块，可以使用本模板中定义的 `codeblock` 命令。其中，`caption` 参数用于指定代码块的标题，`outline` 参数用于指定代码块显示时是否使用边框。下面给出的代码 1.2 是一个简单的 Python 程序。其中的 `<code>` 是这一代码块的标签，意味着这一代码块可以在文档中通过 `@code` 来引用。

代码	渲染结果
<pre>#codeblock(     ```python     def main():         print("Hello, world!")     ```     caption: "一个简单的 Python 程序",     outline: true, ) &lt;code&gt;</pre>	<p>代码 1.2 一个简单的 Python 程序</p> <pre>def main():     print("Hello, world!")</pre>

代码 1.3 中给出了本模板中定义的 codeblock 命令的实现。

代码 1.3 codeblock 命令的实现

```
#let codeblock(raw, caption: none, outline: false) = {
  figure(
    if outline {
      rect(width: 100%)[
        #set align(left)
        #raw
      ]
    } else {
      set align(left)
      raw
    },
    caption: caption, kind: "code", supplement: ""
  )
}
```

## 1.8 参考文献

Typst 支持 BibLaTeX 格式的 `.bib` 文件，同时也新定义了一种基于 YAML 的文献引用格式。要想在文档中引用参考文献，需要在文档中通过调用 `bibliography` 函数来引用参考文献文件。下面是一个示例：

代码	渲染结果
可以像这样引用参考文献： <code>@wang2010guide</code> 和 <code>@kopka2004guide</code> 。  <pre>#bibliography("ref.bib",   style: "ieee" )</pre>	可以像这样引用参考文献：[1] 和 [2]。

注意代码中的 "ref.bib" 也可以是一个数组，比如 ("ref1.bib", "ref2.bib")。

## 第二章 理论

### 2.1 理论一

让我们首先回顾一下第一章中的部分公式：

$$\frac{a^2}{2} \tag{2.1}$$

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \tag{2.2}$$

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \tag{2.3}$$

$$\lim_x = \lim_x \tag{2.4}$$

### 2.2 理论二

在节 2.1 中，我们回顾了第一章中的公式。下面，我们来推导一些新的公式：

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequale doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

### 第三章 展望

目前本模板还有一些不足之处，有待进一步完善：

- 参考文献格式，特别是中文参考文献的格式不完全符合学校有关规定。Discord 上的这个对话显示，Typst 有关功能还在开发中。待有关接口对外开放后，本模板将会进行相应的适配。

### 附录 A 关于 Typst



A.1 在附录中插入图片和公式等

附录中也支持脚注<sup>1</sup>。

附录中也可以插入图片，如图 A.1。

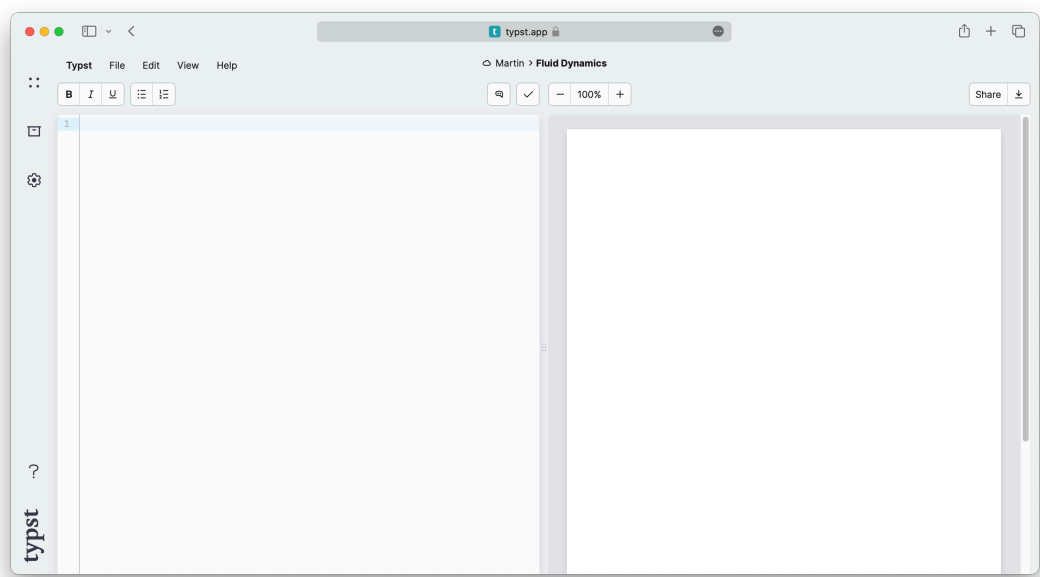


图 A.1 Typst 网页版界面

附录中也可以插入公式，如式 (A.1)。

代码	渲染结果
<code>\$ S = pi r^2 \$ &lt;appendix-eq&gt;</code>	$S = \pi r^2 \tag{A.1}$
<code>\$ mat( 1, 2, ..., 10; 2, 4, ..., 20; 3, 6, ..., 30; dots.v, dots.v, dots.down, dots.v; 10, 20, ..., 100</code>	$\begin{pmatrix} 1 & 2 & \dots & 10 \\ 2 & 4 & \dots & 20 \\ 3 & 6 & \dots & 30 \\ \vdots & \vdots & \ddots & \vdots \\ 10 & 20 & \dots & 100 \end{pmatrix} \tag{A.2}$
	$\mathcal{A} < \mathbb{B} < \mathfrak{C} < \mathsf{D} < \textit{E} < \mathbb{F} \tag{A.3}$
	$\alpha < \beta < \gamma < \delta < \varepsilon \tag{A.4}$

<sup>1</sup>这是一个附录中的脚注。

<code>) \$</code> <code>\$ cal(A) &lt; bb(B) &lt; frac(C) &lt;</code> <code>mono(D) &lt; sans(E) &lt; serif(F) \$</code> <code>\$ bold(alpha &lt; beta &lt; gamma &lt; delta</code> <code>&lt; epsilon) \$</code> <code>\$ upright(zeta &lt; eta &lt; theta &lt; iota</code> <code>&lt; kappa) \$</code> <code>\$ lambda &lt; mu &lt; nu &lt; xi &lt; omicron</code> <code>\$</code> <code>\$ bold(Sigma &lt; Tau) &lt;</code> <code>italic(Upsilon &lt; Phi) &lt; Chi &lt; Psi</code> <code>&lt; Omega \$</code>	$\zeta < \eta < \theta < \iota < \kappa \quad (\text{A.5})$ $\lambda < \mu < \nu < \xi < o \quad (\text{A.6})$ $\Sigma < \mathbf{T} < \Upsilon < \Phi < X < \Psi \quad (\text{A.7})$
--	--

式 (A.8)是一个非常复杂的公式的例子：

代码	渲染结果
<pre>\$ vec(overline(underbracket(underline(1 + 2) + overbrace(3 + dots.c + 10, "large numbers"), underbrace(x + norm(y), y^(w^u) - root(t, z)))), dots.v, u)^(frac(x + 3, y - 2)) \$ &lt;complex&gt;</pre>	<div> <math display="block">\left( \frac{\overbrace{1 + 2 + 3 + \cdots + 10}^{\text{large numbers}}}{\underbrace{\frac{x + \ y\ }{y^{w^u} - \sqrt[t]{z}}}_{u}} \right)^{\frac{x+3}{y-2}} \quad (\text{A.8})</math> </div>

附录中也可以插入代码块，如代码 A.1。

代码 A.1 一个简单的 Rust 程序

```
fn main() {
    println!("Hello, world!");
}
```

## A.2 Typst 的开发者

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

## 附录 B 单独的章节

这是一个单独的章节，目的是为了展示 Typst 的引用功能。

## 参考文献

- [1] 王晓华, 闫其涛, 程智强, 和 张睿, 《科技论文中文摘要写作要点分析》, 编辑学报, 期 S1, 页 53–55, 2010.
- [2] H. Kopka, P. W. Daly, 和 S. Rahtz, *Guide to LATEX*, 卷 4. Addison-Wesley Boston, MA, 2004.

## 致谢

感谢 Typst 开发者的辛勤付出。

## 南方科技大学学位论文原创性声明和使用授权说明

### 原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名

日期：      年      月      日

## 学位论文使用授权说明

(必须装订在提交学校图书馆的印刷本)

本人完全了解南方科技大学关于收集、保存、使用学位论文的规定，即：

- 按照学校要求提交学位论文的印刷本和电子版本；
- 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
- 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
- 因某种特殊原因须要延迟发布学位论文电子版，授权学校 ☐ 一年 / ☐ 两年 / ☐ 三年以后，在校园网上全文发布。

(保密论文在解密后遵守此规定)

论文作者签名

导师签名

日期： 年

月 日