

四川大學

本科生毕业论文（设计）



| | | | |
|--------|----------------------|-----|------|
| 题 目 | 基于 Typst 的川大计院毕业论文模版 | | |
| 学 院 | 计算机学院 | | |
| 专 业 | 计算机科学与技术 | | |
| 学生姓名 | 你的名字 | | |
| 学 号 | 你的学号 | 年 级 | 2021 |
| 指导教师 | 你导 | | |

教务处制表

2025 年 4 月 10 日

基于 Typst 的川大计院毕业论文模版

专业 计算机科学与技术

作者 你的名字 指导老师 你导

[摘要] 先帝创业未半而中道崩殂，今天下三分，益州疲弊，此诚危急存亡之秋也。然侍卫之臣不懈于内，忠志之士忘身于外者，盖追先帝之殊遇，欲报之于陛下也。诚宜开张圣听，以光先帝遗德，恢弘志士之气，不宜妄自菲薄，引喻失义，以塞忠谏之路也。宫中府中，俱为一体；陟罚臧否，不宜异同。若有作奸犯科及为忠善者，宜付有司论其刑赏，以昭陛下平明之理，不宜偏私，使内外异法也。

[主题词] 关键词 1；关键词 2；关键词 3

Abstract

Major Computer Science and Technology

Author Your Name

Mentor Your Mentor

[Abstract] The emperor died before he could finish his work, and the country was divided into three parts. Yizhou was weak and exhausted, and the situation was critical. The ministers of the emperor's court were diligent in their duties, and the loyal and righteous

[Key Words] keyword1; keyword2; keyword3

目 录

| | |
|--------------------|----|
| 1. 绪论 | 1 |
| 1.1. typst 介绍 | 1 |
| 1.2. 基本语法 | 1 |
| 1.2.1. 代码执行 | 1 |
| 1.2.2. 标题 | 1 |
| 2. 本模板相关封装 | 3 |
| 2.1. 图片 | 3 |
| 2.2. 表格 | 3 |
| 2.3. 公式 | 5 |
| 2.4. 列表 | 5 |
| 2.5. 代码 | 6 |
| 2.5.1. 代码块 | 6 |
| 2.5.2. 伪代码 | 6 |
| 3. 其他说明 | 7 |
| 3.1. 文献引用 | 7 |
| 3.2. 致谢部分 | 7 |
| 3.3. 模板相关 | 7 |
| 3.3.1. 数据相关 | 7 |
| 4. 这是一章占位的 | 8 |
| 4.1. 占位的二级标题 1 | 8 |
| 4.2. 占位的二级标题 2 | 8 |
| 4.3. 占位的二级标题 3 | 8 |
| 4.4. 占位的二级标题 4 | 8 |
| 4.4.1. 占位的三级标题 1 | 8 |
| 4.4.2. 占位的三级标题 2 | 8 |
| 4.5. 占位的二级标题 5 | 8 |
| 4.6. 占位的二级标题 6 | 8 |
| 致谢 | 9 |
| 参考文献 | 10 |
| 声明 | 11 |
| 学位论文版权使用授权书 | 11 |

1 绪论

1.1 typst 介绍

typst 是最新最热的标记文本语言，定位与 LaTeX 类似，具有极强的排版能力，通过一定的语法写文档，然后生成 pdf 文件。与 LaTeX 相比有以下的优势：

1. 编译巨快：因为提供增量编译的功能所以在修改后基本能在一秒内编译出 pdf 文件，typst 提供了监听修改自动编译的功能，可以像 Markdown 一样边写边看效果。
2. 环境搭建简单：原生支持中日韩等非拉丁语言，不用再大量折腾字符兼容问题以及下载好几个 G 的环境。只需要下载命令行程序就能开始编译生成 pdf。
3. 语法友好：对于普通的排版需求，上手难度跟 Markdown 相当，同时文本源码阅读性高：不会再充斥一堆反斜杠跟花括号

个人观点：跟 Markdown 一样好用，跟 LaTeX 一样强大



图 1-1 我的 image 实例 0

1.2 基本语法

1.2.1 代码执行

正文可以像前面那样直接写出来，隔行相当于分段。

个人理解：typst 有两种环境，代码和内容，在代码的环境中会按代码去执行，在内容环境中会解析成普通的文本，代码环境用 `{}` 表示，内容环境用 `[]` 表示，在 content 中以 `#` 开头来接上一段代码，比如 `#set rule`，而在花括号包裹的块中调用代码就不需要 `#`。

1.2.2 标题

类似 Markdown 里用 # 表示标题，typst 里用 = 表示标题，一级标题用一个 =，二级标题用两个 =，以此类推。

间距、字体等会自动排版。

#pagebreak() 函数相当于分页符，在华科的要求里，第一级标题应当分页，请手动分页。

2 本模板相关封装

2.1 图片

改用模块 i-figure 实现图片，表格与公式的编号。



图 2-1 我的 image 实例 1

引用的话就在 figure 后加上标签，在原标签前加上前缀 fig:, tbl:, eqt: (分别对应图片，表格与公式)。



图 2-2 我的 image 实例 2

引用 2-1: 图 2-1

2.2 表格

表格跟图片差不多，但是表格的输入要复杂一点，建议在 typst 官网学习，自由度特别高，定制化很强。

同样使用 figure 函数包裹表格

表 2-1 芝士样表



| | Area | Parameters |
|---|-----------------------------|---|
|  | $\pi h \frac{D^2 - d^2}{4}$ | <p>h: height</p> <p>D: outer radius</p> <p>d: inner radius</p> |
|  | $\frac{\sqrt{2}}{12} a^3$ | a : 边长 |

表 2-2 三线表示例

| Country List | Country List | Country List |
|---------------------------|----------------|--------------|
| Country Name or Area Name | ISO ALPHA Code | ISO ALPHA |
| Afghanistan | AF | AFT |
| Aland Islands | AX | ALA |
| Albania | AL | ALB |
| Algeria | DZ | DZA |
| American Samoa | AS | ASM |
| Andorra | AD | AND |
| Angola | AP | AGO |

表 2-3 三线表示例 2

| Country List | Country List | Country List | Country List |
|---------------------------|------------------|--------------|--------------|
| Country Name or Area Name | ISO ALPHA 2 Code | ISO ALPHA 3 | 8 |
| Name | | | |
| Afghanistan | AF | AFT | 7 |
| Aland Islands | AX | ALA | 6 |
| Albania | AL | ALB | 5 |
| Algeria | DZ | DZA | 4 |
| American Samoa | AS | ASM | 3 |
| Andorra | AD | AND | 2 |
| Angola | AP | AGO | 1 |

2.3 公式

公式用两个 \$ 包裹，但是语法跟 LaTeX 并不一样，如果有大量公式需求建议看官网教程 <https://typst.app/docs/reference/math/equation/>。

不需要封装即可为公式编号

$$A = \pi r^2 \tag{2-1}$$

根据公式 (2-1)，推断出公式 (2-2)

$$x < y \Rightarrow x \not\geq y \tag{2-2}$$

然后也有多行的如公式 (2-3)，标签名字可以自定义

$$\begin{aligned} \sum_{k=0}^n k &< 1 + \dots + n \\ &= \frac{n(n+1)}{2} \end{aligned} \tag{2-3}$$

如果不想编号就在公式后面使用标签 <->

$$x < y \Rightarrow x \not\geq y$$

2.4 列表

- 无序列表 1: 1
- 无序列表 2: 2

列表后的正文，应当有缩进。这里加入一个 #indent() 函数来手动生成段落缩进，是因为在目前的 typst 设计里，按英文排版的习惯，连续段落里的第一段不会缩进，也包括各种列表。

1. 有序列表 1
2. 有序列表 2

列表后的正文，应当有缩进，但是这里没有，请自己在段首加上 #indent()

想自己定义可以自己 set numbering，建议用 #[] 包起来保证只在该作用域内生效：

1) 自定义列表 1

a) 自定义列表 1.1

2) 自定义列表 2

a) 自定义列表 2.1

2.5 代码

2.5.1 代码块

```

1  #show "ArtosFlow": name => box[
2    #box(image(
3      "logo.svg",
4      height: 0.7em,
5    ))
6    #name
7  ]
8
9  This report is embedded in the
10 ArtosFlow project. ArtosFlow is a
11 project of the Artos Institute.

```

2.5.2 伪代码

计算机学院建议使用如下的伪代码模版进行伪代码书写。

reference link: <https://typst.app/universe/package/lovelace>

```

      input: integers  $a$  and  $b$ 
output: greatest common divisor of  $a$  and  $b$ 
1      while  $a \neq b$  do
2          if  $a > b$  then
3               $a \leftarrow a - b$ 
4          else
5               $b \leftarrow b - a$ 
6          end
7      end
8      return  $a$ 

```

算法 2-1 The Euclidean algorithm

3 其他说明

3.1 文献引用

引用支持 LaTeX Bib 的格式，也支持更简单好看的 yaml 来配置（尚未流行，推荐优先使用 .bib）在引用时使用 `#bib_cite(<tag>)`，像这样^[1,2]以获得右上的引用标注^[3,4]。

记得在最后加入 `#references("path/to/ref.bib")` 函数的调用来生成参考文献。

由于华科使用自创引用格式，基本上为 GB/T 7714 去掉 [J]、[C]、[M] 刊物类型。Typst 已支持 cs1 自定义参考文献列表，基于这个修改，如果想再自定义新的格式，请修改 `template.typ` 中 `bibliography` 函数中 `style` 参数。

3.2 致谢部分

致谢部分在最后加入 `#acknowledgement()` 函数的调用来生成。

3.3 模板相关

这个模板应该还不是完全版，可能不能覆盖到华科论文的所有 case 要求，如果遇到特殊 case 请提交 issue 说明，或者也可以直接提 pull request

同时，由于 typst 太新了，2023 年 4 月初刚发布 0.1.0 版本，可能会有大的 break change 发生，模板会做相应修改。

主要排版数据参考来自 <https://github.com/zfengg/HUSTtex> 同时有一定肉眼排版成分，所以有可能不完全符合华科排版要求，如果遇到不对的间距、字体等请提交 issue 说明。

3.3.1 数据相关

所有拉丁字母均为 Times New Roman，大小与汉字相同

正文为宋体 12pt，即小四

图表标题为黑体 12pt

表格单元格内容宋体 10.5pt，即五号

一级标题黑体 18pt 加粗，即小二

二级标题 14pt 加粗，即四号

正文行间距 1.24em（肉眼测量，1.5em 与 word 的 1.5 倍行距并不一样）

a4 纸，上下空 2.5cm，左右空 3cm

4 这是一章占位的

4.1 占位的二级标题 1

4.2 占位的二级标题 2

4.3 占位的二级标题 3

4.4 占位的二级标题 4

4.4.1 占位的三级标题 1

4.4.2 占位的三级标题 2

4.5 占位的二级标题 5

4.6 占位的二级标题 6

致 谢

完成本篇论文之际，我要向许多人表达我的感激之情。

首先，我要感谢我的指导教师，他/她对本文提供的宝贵建议和指导。所有这些支持和指导都是无私的，而且让我受益匪浅。

其次，我还要感谢我的家人和朋友们，他们一直以来都是我的支持和鼓励者。他们从未停止鼓舞我，勉励我继续前行，感谢你们一直在我身边，给我幸福和力量。

此外，我还要感谢我的同学们，大家一起度过了很长时间的时光，互相支持和鼓励，共同进步。因为有你们的支持，我才能不断地成长、进步。

最后，我想感谢所有阅读本文的读者，你们的反馈和评价对我非常重要，这也让我意识到了自己写作方面的不足，同时更加明白了自己的研究方向。谢谢大家！

再次向所有支持和鼓励我的人表达我的谢意和感激之情。

本致谢生成自 GitHub Copilot。

参考文献

- [1] Russell Impagliazzo, Ramamohan Paturi, Francis Zane. Which problems have strongly exponential complexity?. Journal of Computer and System Sciences, 2001, 63(4): 512-530
- [2] Sebastian Burckhardt, Manuel Fahndrich, Peli de Halleux, et al. It's Alive! Continuous Feedback in UI Programming. SIGPLAN Not., 2013, 48(6): 95-104
- [3] 刘星, 唐勇. 恶意代码的函数调用图相似性分析. 计算机工程与科学, 2014, 36(3): 481-486
- [4] Christos H. Papadimitriou, Kenneth Steiglitz. Combinatorial optimization: algorithms and complexity. Courier Corporation, 1998

声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得四川大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

本学位论文成果是本人在四川大学读书期间在导师指导下取得的，论文成果归四川大学所有，特此声明。

作者签名：

导师签名：

年 月 日

学位论文版权使用授权书

本学位论文作者完全了解四川大学有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或相关机构送交论文的原件、复印件和电子版，允许论文被查阅和借阅。本人授权四川大学将本学位论文的全部或部分内容编入有关数据库进行信息技术服务，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文，并用于学术活动。

（涉密学位论文在解密后适用于本授权书）

作者签名：

导师签名：

年 月 日