



YanBook 模板

一个现代化的 L^AT_EX 书籍模板



本模板专注于简洁与优雅。灵感来源于现代界面设计，它拥有充足的留白、清晰的排版以及结构化的布局，非常适合用于技术书籍、论文和学术笔记。

作者

yaan

安徽大学

Anhui University

发布日期

December 24, 2025

版本 1.0

一月
只亮
猫想
吃着
了我
我的
的心
奶事
酪

Contents

1 简介	4
1.1 欢迎	4
1.2 中文空格处理	4
1.2.1 基本空格处理	4
1.2.2 使用特殊命令	5
1.2.3 中英文混排的空格	5
1.2.4 调整中文字符间距	5
1.2.5 实用技巧	5
2 功能展示	6
2.1 数学环境	6
2.2 文本块	6
2.3 代码块	7
2.4 图片与子图	8
2.5 表格	8
2.6 高级 TikZ 示例	9
2.6.1 分形 (L-System)	9
2.6.2 复变函数可视化	9
2.6.3 韦恩图	9
2.7 对照块	10
2.8 Mac 风格代码块	10
2.8.1 浅色模式	10
2.8.2 深色模式	11
2.9 算法	11
2.10 新功能: 章节名言、术语表和全页图	11
2.10.1 术语表	12
2.10.2 全页背景	12
3 基础使用指南	14
3.1 项目结构	14
3.2 编译	14
3.2.1 环境要求	14
3.2.2 命令	15
3.3 文档类选项	15

4.2.1	Mac 风格窗口	18
4.3	对照环境	18
4.4	特殊功能	18
4.4.1	章节名言	18
4.4.2	术语表	18
4.4.3	全页背景	19

Nomenclature

\mathbb{R} 实数集

c 真空惯性系中的光速

G 万有引力常数

h 普朗克常数

简介

Mathematics is the queen of the sciences.

— CARL FRIEDRICH GAUSS

1.1 欢迎

这是一个使用 `yanbook` 文档类的示例文档。

Definition 1.1.1 ▶ Yanbook

`Yanbook` 是一个基于 `amznotes` 的自定义 LaTeX 文档类。

Theorem 1.1.2 ▶ 示理定理

这是一个定理盒子。

$$E = mc^2$$

1.2 中文空格处理

在 LaTeX 中处理中文时，空格的处理方式可能会让人困惑。以下是几种在中文中添加空格的方法：

1.2.1 基本空格处理

在中文文本中，直接输入空格通常不会显示，因为 CJK 字符（中文、日文、韩文）的排版规则与西文不同。例如：

这是没有空格的中文句子。

1.2.2 使用特殊命令

要插入空格，您可以使用以下方法：

1. 使用 \ 命令：这里有空格
2. 使用 ~ 命令：这里有不间断空格
3. 使用 \hspace{} 命令：这里有指定宽度的空格
4. 使用 \quad 或 \quad：这里有更大的空格

1.2.3 中英文混排的空格

在中英文混排时，LaTeX 通常会自动处理空格：

这是一个包含 English 和 数字 123 的句子。

1.2.4 调整中文字符间距

如果需要调整中文字符之间的间距，可以临时使用：

```
\begin{CJKspace}  
这里 每个 字 之 间 会 有 空 格  
\end{CJKspace}
```

但请注意，这不是标准的中文排版方式，一般情况下不推荐使用。

1.2.5 实用技巧

- 在需要强调分隔的地方，使用标点符号而非空格
- 使用引号、括号等标点符号来组织文本结构
- 在中英文混排时，适当在英文前后添加空格以提高可读性

功能展示

2.1 数学环境

我们可以使用标准的 LaTeX 数学环境。

行内公式: $E = mc^2$ 。

行间公式:

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

Theorem 2.1.1 ▶ 微积分基本定理

设 f 是定义在闭区间 $[a, b]$ 上的连续实值函数。设 F 是定义在 $[a, b]$ 上的函数, 对于所有 $x \in [a, b]$, 有

$$F(x) = \int_a^x f(t) dt$$

则 F 在 $[a, b]$ 上一致连续, 在开区间 (a, b) 上可导, 并且对于所有 $x \in (a, b)$, 有

$$F'(x) = f(x)$$

Lemma 2.1.2 ▶ 夹逼定理

如果 $g(x) \leq f(x) \leq h(x)$ 且 $\lim_{x \rightarrow a} g(x) = \lim_{x \rightarrow a} h(x) = L$, 那么 $\lim_{x \rightarrow a} f(x) = L$ 。

2.2 文本块

Definition 2.2.1 ▶ 极限的定义

设 f 是定义在包含 c 的开区间 (可能除去 c 点) 上的函数, 且 L 是一个实数。语句

$$\lim_{x \rightarrow c} f(x) = L$$

意味着对于任意 $\epsilon > 0$, 存在一个 $\delta > 0$, 使得如果 $0 < |x - c| < \delta$, 则 $|f(x) - L| < \epsilon$ 。

Example 2.2.2 ▶ 计算示例

计算 $x \rightarrow 2$ 时 $f(x) = x^2$ 的极限。

解：因为 $f(x)$ 是连续的，我们直接代入数值：

$$\lim_{x \rightarrow 2} x^2 = 2^2 = 4$$

Technique 2.2.3 ▶ 分部积分法

分部积分公式为：

$$\int u \, dv = uv - \int v \, du$$

当被积函数是两个函数的乘积时非常有用。

这是一个简单的提示框。用于强调重要信息或警告。

通用信息

这是一个通用盒子。可用于任何需要与正文区分的内容。

2.3 代码块

由于 `code` 选项当前已禁用（需要 Python 和 Pygments），这里展示一个使用盒子包裹 `verbatim` 的模拟代码块。

Python 示例

```
def hello_world():
    print("Hello, World!")
    return True

if __name__ == "__main__":
    hello_world()
```

如果你在 `documentclass` 中启用了 `code` 选项，你可以使用 `amzcode` 环境进行语法高亮：

```
\begin{amzcode}{python}
```

```
def hello_world():
    print("Hello, World!")
\end{amzcode}
```

2.4 图片与子图

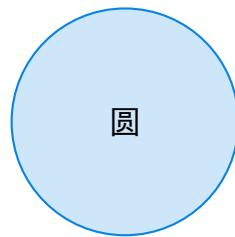
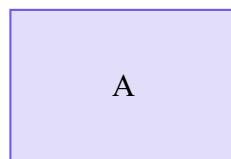
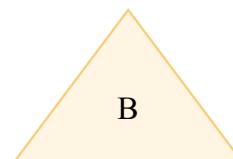


Figure 2.1: 使用 TikZ 绘制的示例图片

我们也可以并排排列图片。



(a) 子图 A



(b) 子图 B

Figure 2.2: 并排的两个子图

2.5 表格

这是一个使用 `booktabs` 的专业三线表示例。

Table 2.1: 模型对比

模型	准确率	时间 (ms)
基准模型	85.4%	12.5
模型 A	88.2%	15.3
模型 B	91.7%	22.1

2.6 高级 TikZ 示例

2.6.1 分形 (L-System)

这是一个使用 Lindenmayer 系统生成的植物分形。



Figure 2.3: 植物分形

2.6.2 复变函数可视化

这是一个使用 `pgfplots` 绘制的函数 $z = xe^{-x^2-y^2}$ 的 3D 图。

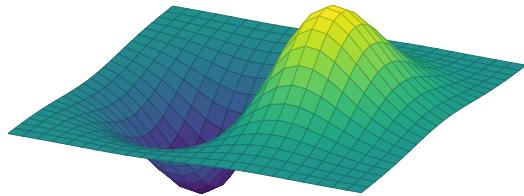


Figure 2.4: 3D 曲面图

2.6.3 韦恩图

表示三个集合 A 、 B 和 C 交集的韦恩图。

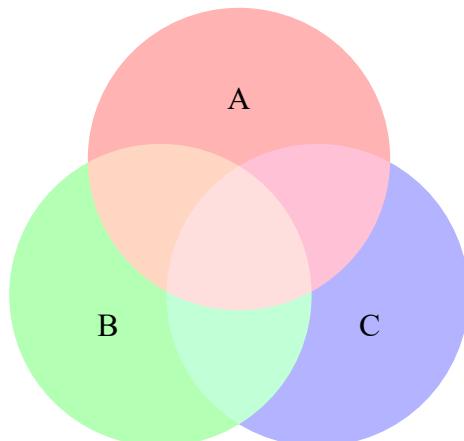


Figure 2.5: 三个集合的韦恩图

2.7 对照块

我们可以使用 `comparison` 环境并排对比两个概念。

牛顿力学

- 适用于宏观物体。
- 速度远小于光速。
- 空间和时间是绝对的。
- $F = ma$

相对论力学

- 适用于所有速度。
- 接近光速时必须使用。
- 空间和时间是相对的。
- $E = mc^2$

2.8 Mac 风格代码块

我们提供两种风格的代码块：浅色和深色。

2.8.1 浅色模式

hello.py

```
1 def factorial(n):
2     """Calculates the factorial of n."""
3     if n == 0:
4         return 1
5     else:
6         return n * factorial(n-1)
7
8 print(factorial(5))
```

2.8.2 深色模式

```
1 #include <iostream>
2
3 int main() {
4     // This is a dark mode code block
5     std::cout << "Hello, World!" << std::endl;
6     return 0;
7 }
```

2.9 算法

我们也可以精美地排版算法。

Algorithm: 欧几里得算法

```
1: procedure EUCLID( $a, b$ )                               ▷  $a$  和  $b$  的最大公约数
2:    $r \leftarrow a \bmod b$ 
3:   while  $r \neq 0$  do                                        ▷ 如果  $r$  为 0 则得到结果
4:      $a \leftarrow b$ 
5:      $b \leftarrow r$ 
6:      $r \leftarrow a \bmod b$ 
7:   end while
8:   return  $b$                                                 ▷ 最大公约数是  $b$ 
9: end procedure
```

2.10 新功能：章节名言、术语表和全页图

We must know. We will know.

— DAVID HILBERT

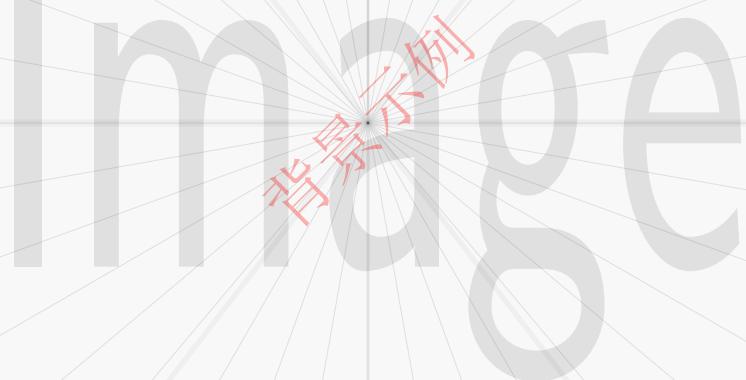
2.10.1 术语表

我们在这里定义一些符号。

符号 c 、 h 和 G 是基本常数。 \mathbb{R} 是一个集合。请查看文档开头的术语表。

2.10.2 全页背景

下一页将有一个全页背景图（本例中使用 TikZ 模拟）。



Image

基础使用指南

The essence of mathematics lies in its freedom.

— GEORG CANTOR

3.1 项目结构

yanbook 模板的组织结构如下：

- `main.tex`: 主文档文件。
- `yanbook.cls`: 定义样式的自定义类文件。
- `Makefile`: 构建自动化脚本。
- `chapters/`: 包含章节内容文件的文件夹。
- `contents/`: 包含封面等其他内容的文件夹。
- `images/`: 用于存储图片的文件夹。
- `output/`: 生成的 PDF 存放目录。
- `build/`: 中间构建文件目录。

3.2 编译

本项目使用 `Makefile` 来简化编译过程。

3.2.1 环境要求

- TeX 发行版 (TeX Live, MiKTeX, 或 MacTeX)
- XeLaTeX 引擎 (用于 Unicode 和字体支持)
- Make 工具 (可选但推荐)

3.2.2 命令

要编译项目，请在项目根目录打开终端并运行：

bash

```
1 make
```

该命令将：

1. 使用 `xelatex` 编译文档。
2. 运行 `biber` 处理参考文献(如果需要)。
3. 运行 `makeindex` 处理术语表。
4. 重新编译以解决引用。
5. 将最终的 PDF 移动到 `output/` 目录。

清理构建文件：

bash

```
1 make clean
```

3.3 文档类选项

`yanbook` 类支持在 `main.tex` 中传递几个选项：

latex

```
1 \documentclass[math, code, fastcompile]{yanbook}
```

math 启用数学相关环境 (`thmbox`, `lembox`) 并加载 `mathtools`, `amsthm`, `derivative` 等宏包。

code 启用 `minted` 宏包以进行高级代码高亮(需要安装 Python 和 Pygments)。注意：模板也包含不需要此选项的独立 Mac 风格代码块。

fastcompile 禁用一些耗时的样式功能(如花哨的章节标题和部分盒子阴影)以加快草稿阶段的编译速度。

高级功能详解

4.1 文本盒子

A mathematician is a machine for turning coffee into theorems.

— PAUL ERDŐS

模板提供了几种预定义的彩色盒子，用于不同类型的内容。

4.1.1 定义与定理

```

1 \begin{dfnbox}{标题}{标签}
2   内容...
3 \end{dfnbox}

4
5 \begin{thmbox}{标题}{标签}
6   内容...
7 \end{thmbox}
```

可用盒子：

- **dfnbox**: 定义 (蓝色)
- **thmbox**: 定理 (紫色, 需要 **math** 选项)
- **lembox**: 引理 (紫色, 需要 **math** 选项)
- **exbox**: 示例 (黄色)
- **tecbox**: 技巧 (橙色)
- **genbox**: 通用信息 (绿色)
- **notebox**: 注意/警告 (红色, 简单风格)

4.2 代码块

4.2.1 Mac 风格窗口

这些块模仿 macOS 窗口的外观。它们使用 `listings` 宏包，不需要 Python。

浅色模式：

```
1 \begin{maccode}{python}
2 def hello():
3     print("Hello")
4 \end{maccode}
```

深色模式：

```
1 \begin{macdarkcode}{python}
2 def hello():
3     print("Hello")
4 \end{macdarkcode}
```

4.3 对照环境

使用 `comparison` 环境并排显示两列内容，中间有垂直分隔线。

```
1 \begin{comparison}
2 左侧内容
3 \tcblower
4 右侧内容
5 \end{comparison}
```

4.4 特殊功能

4.4.1 章节名言

在章节开头添加名言：

```
1 \epigraph{名言内容}{作者}
```

4.4.2 术语表

为术语表定义符号：

```
1 \nomenclature{$x$}{x 的描述}
```

列表使用 `\printnomenclature` 打印。

4.4.3 全页背景

在当前页面插入全页背景图片：

```
1 \fullpagebg[0.5]{path/to/image.jpg}
```

可选参数控制不透明度 (默认为 1)。