

@ 算法时空

目录

摘要

简单明了.

关键词: X; Y; Z

ABSTRACT

Clear & Simple.

Keywords: X; Y; Z

第1章 结构

1.1 节

1.1.1 小节

第2章 文字

2.1 举例

书写, 用心.
段之间以空行分隔.

2.2 代码抄录

可以直接列出代码(设定缩进距离为2, 每行编号并从0开始):

```
0  for (int i = 0; i < n; ++i)
1    for (int j = 0; j < n; ++j)
2      cout << i * j << ' ';
```

也可以使用`VerbatimInput`命令直接包含代码文件。

代码变量是`i`, 复杂的是`int i = 0`。

第3章 公式

最常见的是行内公式, 例如集合 X 中的元素 a , 初学者容易直接使用普通符号而不是数学模式. 另外一个问题是斜体, 最好使用斜体方式.

行间公式一般带编号, 我们可使用 `label` 引用, 例如公式??给出了渐近记号.

$$\log(n!) = \Theta(n \log n) \quad (3.1)$$

注意写一段就编译一段, 这样容易查错, 特别是公式.

3.1 初学L^AT_EX常见的数学符号误用

我们给出几个典型例子(括号内附有代码).

- 是 $\log x$ (`\log{x}`) 而不是 $\log x$ (`\log x`);
- 是 \min (`\min`) 和 \max (`\max`) 而不是 \min (`min`) 和 \max (`max`);
- 是 \Pr (`\Pr`) 而不是 Pr (`Pr`);
- 是 $\sin x$ (`\sin{x}`) 和 $\cos x$ (`\cos{x}`) 而不是 $\sin x$ (`\sin x`) 和 $\cos x$ (`\cos{x}`);
- 是 $x \times y$ (`x \times y`) 而不是 $x * y$ (`x * y`);
- 是 \arg (`\arg`) 而不是 arg (`arg`);

第4章 图片

一般采用浮动的图，以选项 `[!htbp]` 标记，如图??所示，注意其中还有两个子图??和??.



图 4.1 主图

如果有特殊的需要，可以用固定位置的图片，在后面加上 `[h]` 选项即可，例如图??.



图 4.2 算法三部曲

当然也不一定能够完全在当前位置，可能当前位置不够会挤到下一页。

不要用屏幕截图，实验结果可以用软件的导出图，¹ 数据可以导出文本利用表格或者抄录形式。

¹得用PNG格式，其他的展示型图像可以用JPG格式。

第5章 表格

一般采用浮动的表格, 以选项[!htbp]标记, 注意表格不能写“如下所示”要写“如??所示”. 表??使用了单元格不同位置的标记(例如c/1/r标记).

	Books	
1	Introduction to Algorithms	3
2	The Algorithm Design Manual	2
3	Algorithms	4

表 5.1 算法三部曲

表??下面还有子表??和??, 注意命名不同.

表 5.2 主表

0	1	2	3
4	5	6	7

(a) 子表(十进制)

000	001	010	011
100	101	110	111

(b) 子表(二进制)

最好要把每一页填满, 这样排版问题会少很多。

第6章 引用

最常见的引用是参考文献的引用. 例如参考文献[?]的标记为SX(也即`\bibitem{SX}`), 于是我们可以使用`\cite{SX}`实现引用参考文献[?]的效果, 而这本书便是使用 \LaTeX 排版.

引用需要排次序, 所以需要两次编译, 第1次识别所有的标记并编号, 第2次将编号填入并显示.

其他引用一般使用`label`给出标记, 再用`ref`命令引用. 例如以`\label{eq:log_factorial}`给出公式标记`eq:log_factorial`, 再用`\ref{eq:log_factorial}`引用得到`??`. 为了区别不同的标记, 可用前缀配合冒号区分: 公式的前缀`eq`, 图的前缀`fig`, 表的前缀`tab`. 列举如下:

- 公式标记为`eq:log_factorial`, 也即公式`??`.
- 图标记为`fig:books`, 也即图`??`.
- 表标记为`tab:trilogy`, 也即表`??`.

第7章 版式

我们在`NCL.def`引入的版式设计非常少, 如果需要调整可改动该文件, 例如更改页眉的九章论文, 或者公式或者图表的编号格式.

如果每章首页不需要任何页眉页脚, 可使用我们定义的`echapter`命令. 当然还有其他方法: <https://tex.stackexchange.com/questions/19738/why-doesnt-pagestyleempty-work-on-the-first-page-of-a-chapter>.

另外, `thesis.tex`和`abstract.tex`也有少量排版设定, 可酌情增减.

第8章 辅助

如今有很多辅助工具可以帮助我们更好地完成 \LaTeX 文档.

8.1 Typora

Typora是一个“所见即所得”Markdown编辑器(<https://typora.io>), 特别是很好地配合MathJax显示 \LaTeX 公式. 平时可以先用这个软件写一点片段, 若能正确展示, 再复制到 \TeX 文件中, 可以提高文本编写效率.

8.2 Microsoft Math Solver

Microsoft Math Solver(“微软数学”)这款APP的主要功能是求解数学问题, 但是我们可以用来处理复杂的公式, 在平板上手写识别后可以复制 \LaTeX 源代码.

8.3 表格自动转换

对于数据量较大的表格, 可以寻找转换工具(也有很多在线版本), 会节约很多时间.
另外, 实验数据最好由编程语言提供的软件包转换.

第9章 进阶

致谢

感谢Donald E. Knuth和Leslie Lamport, 感谢 $\text{T}_{\text{E}}\text{X}$ 和 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

参考文献

- [1] Steven S. Skiena [著], 谢颢 [译]. 算法设计指南 (第2版) [M]. 北京: 清华大学出版社, 2017.