



数据结构与算法

Data Structure and Algorithm

极夜酱

目录

I	数据结构	1
1	数据结构与算法	2
1.1	算法	2

Part I

数据结构

Chapter 1 数据结构与算法

1.1 算法

1.1.1 算法 (Algorithm)

算法是一个很古老的概念，最早来自数学领域。

有一个关于算法的小故事：在很久很久以前，曾经有一个顽皮又聪明的熊孩子，天天在课堂上调皮捣蛋。终于有一天，老师忍无可忍，对熊孩子说：



臭小子，你又调皮啊！今天罚你算加法，算出

$1+2+3+4+5+6+7+\dots$ 一直加到10000的结果，算不完不许回家！



嘿嘿，我算就是了。

老师以为，熊孩子会按部就班地一步一步计算：

$$1 + 2 = 3$$

$$3 + 3 = 6$$

$$6 + 4 = 10$$

$$10 + 5 = 15$$

...

这还不得算到明天天亮？够这小子受的！老师心里幸灾乐祸地想着。谁知仅仅几分钟后……



老师，我算完了！结果是50 005 000，对不对？



这、这、这……你小子怎么算得这么快？我读书多，你骗

不了我的！

看着老师惊讶的表情，熊孩子微微一笑，讲出了他的计算方法。

首先把从 1 到 10000 这 10000 个数字两两分组相加：

$$1 + 10000 = 10001$$

$$2 + 9999 = 10001$$

$$3 + 9998 = 10001$$

$$4 + 9997 = 10001$$

...

一共有 $10000 \div 2 = 5000$ 组，所以 1 到 10000 相加的总和可以这样来计算：

$$(1 + 10000) \times 10000 \div 2 = 50005000$$

这个熊孩子就是后来著名的犹太数学家约翰·卡尔·弗里德里希·高斯，而他所采用的这种等差数列求和的方法，被称为高斯算法。



图 1.1: 约翰·卡尔·弗里德里希·高斯

算法是解决问题的一种方法或一个过程，是一个由若干运算或指令组成的有穷序列。求解问题的算法可以看作是输入实例与输出之间的函数。

算法有 5 个特点：

1. 有穷性 (finiteness)：算法必须能在执行有限个步骤之后终止。
2. 确定性 (definiteness)：算法的每一步骤必须有确切的定义。
3. 输入项 (input)：一个算法有 0 个或多个输入。
4. 输出项 (output)：一个算法有一个或多个输出，没有输出的算法是毫无意义的。
5. 可行性 (effectiveness)：算法中执行的任何计算步骤都是可以被分解为基本的可执行的操作步。

1.1.2 算法描述

算法是可完成特定任务的一系列步骤，算法的计算过程定义明确，通过一些值作为输入并产生一些值作为输出。

流程图 (flow chart) 是算法的一种图形化表示方式，使用一组预定义的符号来说明如何执行特定任务。

- 圆角矩形：开始和结束
- 矩形：数据处理
- 平行四边形：输入/输出
- 菱形：分支判断条件
- 流程线：步骤

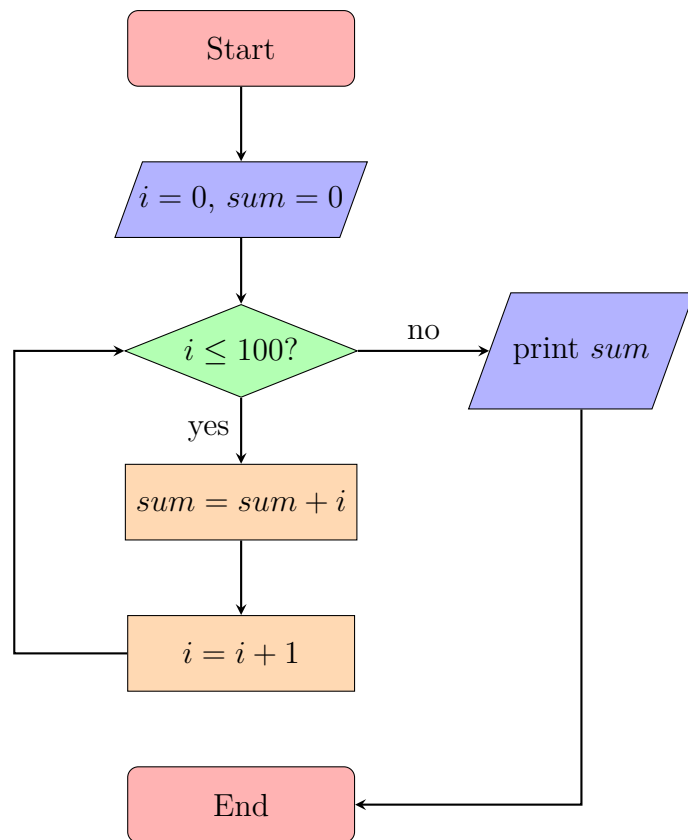


图 1.2: 计算 $\sum_{i=1}^{100} i$ 的流程图

伪代码 (pseudocode) 是一种非正式的，类似于英语结构的，用于描述模块结构图的语言。使用伪代码的目的是使被描述的算法可以容易地以任何一种编程语言实现。