

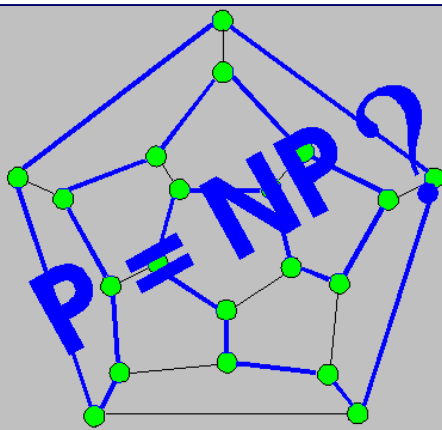


1-算法概述

陆伟

算法设计与分析

Introduction to the Design and Analysis of
Algorithms



September 18, 2022

Lecture Overview

1

- 算法基本概念

2

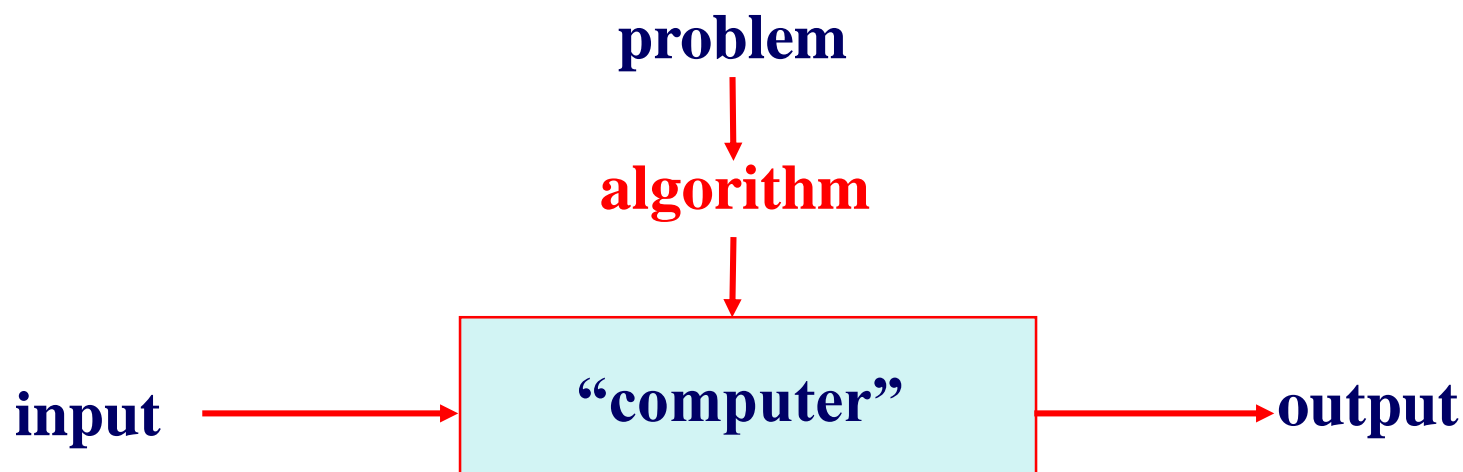
- 算法求解问题基本过程

3

- 常见问题类型

1. 算法基本概念

- **算法**是一系列解决问题的清晰指令，也就是说，对于符合一定规范的输入，算法能够在有限时间内获得所要求的输出。
- **算法**是解决问题的一种方法或过程，它是由若干条指令组成的有穷序列。



1. 算法基本概念

- 一个算法应该具有以下几个方面的特征：
 - **输入**：有零或多个外部量作为算法的输入。
 - **输出**：算法产生至少一个量作为输出。
 - **确定性**：组成算法的每条指令清晰、无歧义。
 - **有效性**：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。
 - **有限性**：算法中每条指令的执行次数有限，执行每条指令的时间也有限。
- 算法与程序区别

1.算法基本概念-案例分析

问题： 在给定序列中查找某个元素。

例如：

给定序列： 3， 9， 7， 8， 5， 12， 15

搜索8

1.算法基本概念-案例分析

思路：顺序扫描，逐个对比。

//输入：给定元素序列a，待查找元素x

//输出：若找到元素x,返回x在a中的下标位置，否则返回-1

sequentialSearch(a, x)

第一步：初始化下标变量i指向序列a的第一个元素；

第二步：比较a[i]和x；

第三步：若a[i]等于x，返回i值；否则进入第四步；

第四步：i值加1，若i增量小于a的长度返回第二步；否则返回-1。

1.算法基本概念-案例分析

思路：二分查找。

递归？
非递归？

//输入：给定有序元素序列a，待查找元素x

//输出：若找到元素x,返回x在a中的下标位置，否则返回-1

binarySearch(a[], x)

$l \leftarrow 0, r \leftarrow a.length$

while($l \leq r$)

$m = (l + r) / 2$

if($x < a[m]$) $r \leftarrow m - 1$

if($x > a[m]$) $l \leftarrow m + 1$

if($x = a[m]$) return m

return -1

若序列中有重复元素，如何？

1. 算法基本概念-案例分析

问题：求两个正整数 m 和 n 的最大公约数 $\gcd(m, n)$ 。

连续整数检测法

更相减损法

质因数分解法

欧几里得算法

1. 算法基本概念-案例分析

欧几里得算法

//输入：两个正整数m,n

//输出：m, n的最大公约数

euclid(m,n)

if(m%n = 0) return n

if(n%m = 0) return m

if(m>=n) euclid(n, m%n)

else euclid(m, n%m)

1.算法基本概念-案例分析

质因数分解法

//输入：两个正整数 m, n

//输出： m, n 的最大公约数

计算 $\gcd(m, n)$ 的质因数分解算法

第一步：找到 m 的所有质因数。

第二步：找到 n 的所有质因数。

第三步：从第一步和第二步中求得的质因数分解式找出所有的公因数。

第四步：将第三步中找的质因数相乘，其结果作为给定数字的最大公约数。

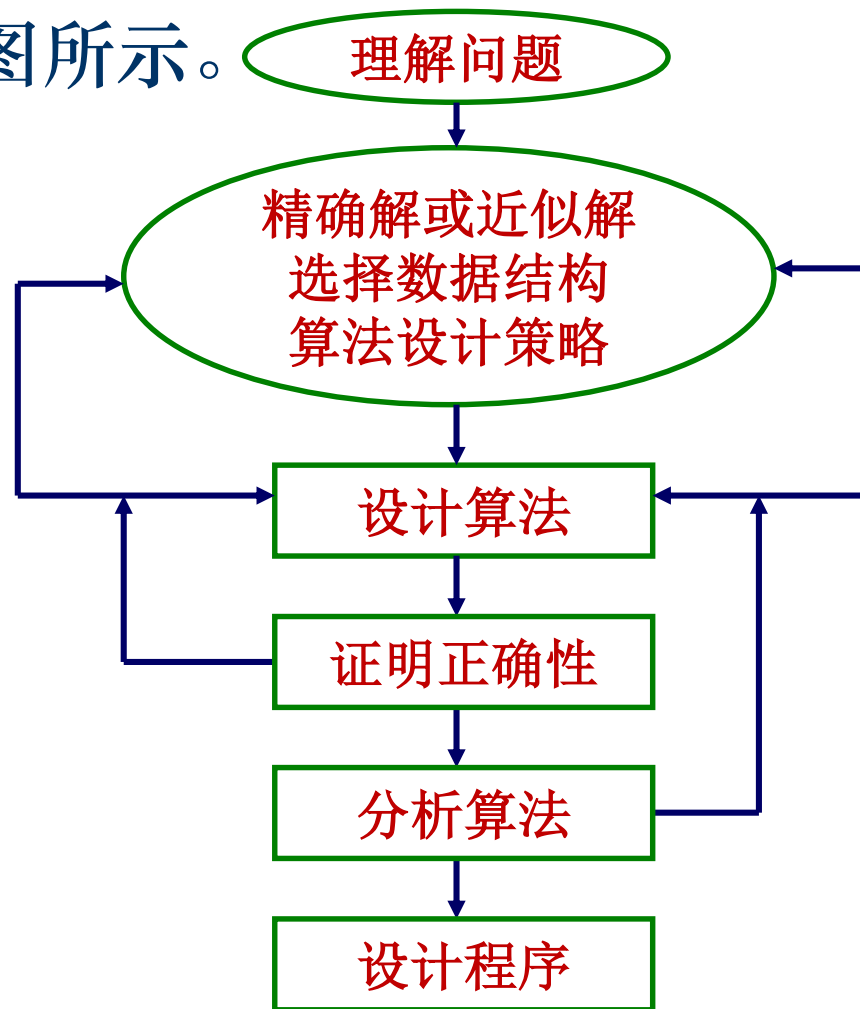
2. 算法求解问题基本过程

- 算法对于精确定义的结构化过程的强调，使得计算机科学有别于其他学科，特别是理论数学。
- 理论数学一般仅满足于证明某个问题是否有解，或者对解的性质进行研究。算法关注如何获得问题的解。
- 例如：幻方问题

2	9	4
7	5	3
6	1	8

2. 算法求解问题基本过程

- 一般情况下，算法的设计与分析过程中经历的典型步骤如图所示。



2. 算法求解问题基本过程

■ 理解问题

- 手工处理小规模问题可以帮助理解
- 确定算法需要处理的实例范围
- 对问题进行抽象，用数学的方法加以重新表述

问题：有 n 个客户带来 n 项任务等待在一个机器上加工，每个任务都需要一定的加工时间，为了使尽可能多的客户满意，希望找到使得总等待时间最少的任务加工调度方案。

给定任务集合 $S = \{1, 2, \dots, n\}$ ， $\forall j \in S$ ，其加工时间 $t_j \in \mathbb{Z}^+$ ，求 S 的一个排列 i_1, i_2, \dots, i_n ，使得

$$\sum_{k=1}^n (n - k + 1)t_{i_k}$$

2.算法求解问题基本过程

- 了解计算设备性能
 - 类冯·诺依曼机器—随机存取(RAM)，指令顺序执行
 - 并行计算
- 在精确解法与近似解法之间选择
- 确定适当的数据结构

2. 算法求解问题基本过程

■ 算法设计

- 算法思想
- 算法描述

- -自然语言, -流程图, -伪代码

■ 算法的正确性证明

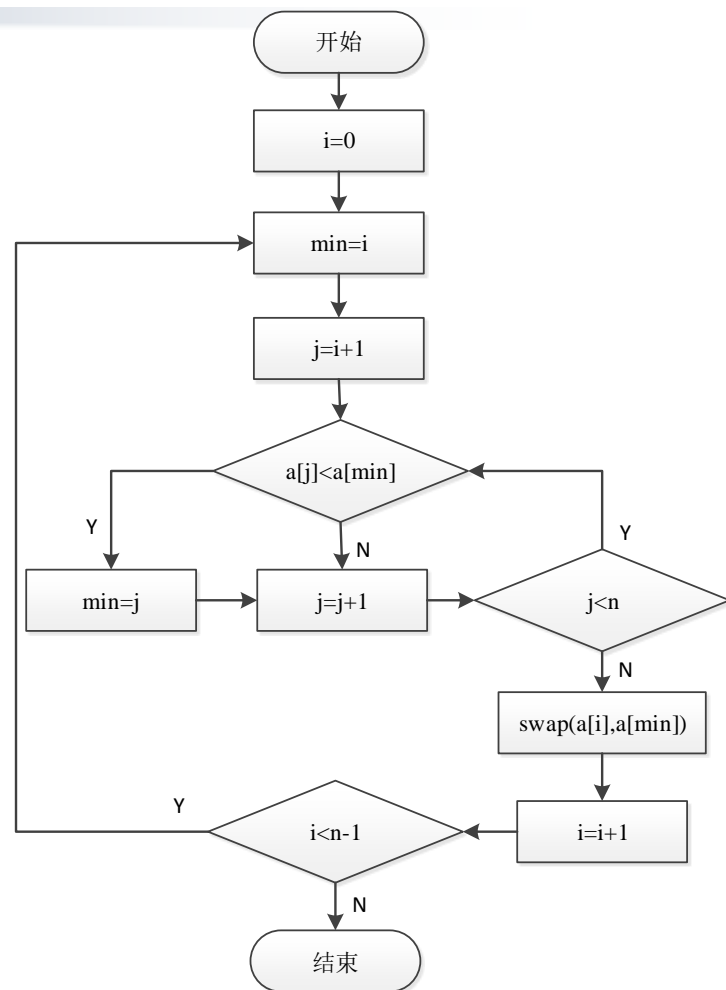
- 归纳法, 反证法

■ 算法分析

- 正确性, 效率, 简单性, 一般性

■ 算法的程序实现

- 程序的正确性问题
- Spec, Test-first Programming



2. 算法求解问题基本过程

追求更好和完美是一件好事，但完美的代价往往是高昂的，我们不应该总是提倡。当设计与实现算法是一种工程行为时，我们需要在**资源有限**的情况下，在**互斥的目标**之间做权衡，设计者的时间也是这样一种资源。

虽然实际计算中，绝大多数问题都能用算法解决，但我们不能强求算法，因为并不是每个问题都能够用算法的方法来解决。

3.常见问题类型

- 排序
- 查找
- 字符串处理
- 图的问题
- 组合问题
- 几何问题
- 数值问题
- 方程问题

Discuss

