

## 第一章 算法在计算中的作用

1. **算法**(algorithm)可以用英语、计算机程序甚至是硬件设计来表达，它是一系列的计算步骤，用来**将输入数据转换成输出结果**。简单的说，**算法是定义良好的计算过程**。
2. **算法**有正确的，也有不正确的，如果不正确算法的错误率可以得到控制的话，它们有时也是有用的，但**一般我们只关注正确的算法**。
3. **数据结构**是存储和组织数据的一种方式，以便对数据进行访问和修改。没有一种数据结构可以适用于所有用途和目的，因此了解数据结构的长处和局限性相当重要。
4. 衡量算法效率的常用标准是**速度**。

### 【练习】

#### 1.1-1 :

- a) 排序：将一次考试中 500 名考生的成绩按分数从高到低进行排名。
- b) 确定多矩阵相乘的最佳顺序：某实验模型，需要计算 $A_1 \times A_2 \times A_3 \cdots \times A_{17}$ 这 17 个矩阵的积，根据矩阵乘法的结合律确定计算顺序，以达到计算乘法次数最少的目的。  
✖矩阵乘法的结合律： $(A \times B) \times C = A \times (B \times C)$
- c) 找出凸壳：木板上钉了 21 个钉子，以其中一些钉子为顶点组成的凸多边形可以包含所有 21 个钉子，找出使凸多边形达到最小的所有钉子。

#### 1.1-2 :

占用资源大小，问题的解决程度，答案精度等。

#### 1.1-3 :

栈 ( LIFO )：长处是可以严格按照后进先出顺序，非常适合如保存程序调用的返回地址之类的特殊应用，缺点是无法进行随机读写。

#### 1.1-4 :

相似之处：都是寻求最短的路线

不同之处：最短路线问题是从若干条可选线路中选择一条线路使之在两个点之间达到

交通路径最短，不需要经过所有的点；旅行商人问题中如果把仓库看做是图中的一个点的话，首先要满足遍历所有的点，然后在所有满足的线路中选择最短的线路，其复杂程度要高于最短路径。

✖给定一幅道路交通图，上面标注除了每一对相邻交叉路口之间的距离。我们的目的就是确定一个交叉路口到另一个交叉路口之间的最短路线。即使不允许每一条路线自我交叉，可能的路线数量也会是巨大的。在所有可能的路线中，如何来选出最短的路线？【最短路径问题】

✖假设有一个火车运输公司，它有一个中央仓库。每一天，它都要在仓库中将货物装满火车，并让它驶往若干个地方去送货。在一天结束时，这辆火车必须最后回到仓库，以便第二天再装货物。为了降低成本，该公司希望选择一条送火车行驶距离最短的送货顺序。【旅行商人问题】

### 1.1-5：

A．公司需经常往某地大量派人出差，该地有 600 多家旅馆，其中可以满足公司员工住宿要求的有 390 家，假设该 390 家旅馆的住宿费用不全相同，且价格起伏较大。公司为某次出差员工指定旅馆的标准：在出差办公地 30 分钟车程内，选择价格最低的旅馆，如果价格一样，取车程更近的那家。为了最大化的节省公司运营成本，此例只能用最佳方案来解决，并且总是可以找到最佳方案。

B．在 A 例子中，假设在出差办公地 30 分钟车程内，满足公司员工住宿要求的旅馆有很多家，且最高价格与最低价格相差较小，在 1%以内，那么可以不管价格，选择离出差办公地较近、地段较好的任意一家旅馆住宿即可，这个近似最优解足以满足公司节省运行成本的目的。■

5. 计算机可以做的很快，但还不能是无限快。存储器可以做到很便宜，但不会是免费的。因此，计算时间是一种有限的资源，存储空间也是有种有限的资源。这些有限的资源必须被有效的使用，那些时间和空间上有效的算法就有助于做到这一点。

6. 算法就像计算机硬件一样，**是一种技术**。总体的系统性能不仅依赖于选择快速的硬件，还依赖于选择有效的算法。算法是当代计算机中用到的大部分技术的核心。

**例：**一台性能较高的计算机 A(每秒执行 10 亿条指令)和另一台性能较低的计算机 B (每秒执行 1000 万条指令)，分别在 A、B 上使用插入排序 (Insertion Sort)、合并排序 (Merge Sort) 算法对 100 万个数进行排序。假设世界上最能干的程序员采用机器语言，来为计算机编写插入排序算法的代码，所得到的代码需要  $2n^2$  条指令来排序  $n$  个数 (此处， $c_1 = 2$ )。另一方面，让一位平均熟练水平的程序员，采用某种低效编译器的高级语言来为计算机 B 编写合并排序算法的代码，所得到的代码有  $50n \lg n$  条指令 (这里  $c_2 = 50$ )。

为了排序这 100 万个数，计算机 A 花费时间：

$$\frac{2 \times (10^6)^2}{10^9} = 2000s$$

计算机 B 花费时间：

$$\frac{50 \times 10^6 \times \lg 10^6}{10^7} \approx 100s$$

计算机 B 由于采用了一个运行时间**增长的更为缓慢**的算法，运行速度大大快于计算机 A。

### 【练习】

#### 1.2-1：

在一次有 210 人参加的考试结束后，老师需要对平均分进行计算。假设老师不借助

于计算机类的自动工具，而是采取传统的手工计算平均分。

计算方法：看过考分后，大概估计一个值 $K$ ，然后与 210 个分数的数据进行比较，并且记下分数与估计值 $K$ 的误差 $\Delta_i (i = 1, 2, 3, \dots, 210)$ 。在所有的 $\Delta$ 记录下后，检查这些数据，如果发现有 $\Delta_i = -\Delta_j$ ，则把这一对 $\Delta$ 都划去，不作计算。假设最后计算出来的这些误差总和是 $\Sigma$ ，那么最后的平均分：

$$\bar{N} = K + \frac{\Sigma}{210}$$

算法的功能：该算法对于计算机这样的自动化计算工具来说意义不大，但是对于手工计算来说，却是很实用的。在计算的过程中，首先把大数转换成了小数，然后在比较时又直接把一些数据剔除，减少了运算量。最终大大加快了计算效率，并且提高了计算准确度。

$$1.2-2 : 8n^2 < 64n \lg n \implies 2 \leq n \leq 43$$

$$1.2-3 : 100n^2 < 2^n \implies n_{\min} = 15 \blacksquare$$

### 【思考题】

	1 秒	1 分钟	1 小时	1 天	1 个月	1 年	1 个世纪
$\lg n$	$2^{10^6}$	$2^{6 \times 10^7}$	$2^{3.6 \times 10^9}$	$2^{8.64 \times 10^{10}}$	$2^{2.592 \times 10^{12}}$	$2^{9.4608 \times 10^{14}}$	$2^{9.4608 \times 10^{16}}$
$\sqrt{n}$	$10^{12}$	$3.6 \times 10^{15}$	$1.296 \times 10^{19}$	$7.46496 \times 10^{21}$	$6.718464 \times 10^{24}$	$8.950673664 \times 10^{29}$	$8.950673664 \times 10^{33}$
$n$	$10^6$	$6 \times 10^7$	$3.6 \times 10^9$	$8.64 \times 10^{10}$	$2.592 \times 10^{12}$	$9.4608 \times 10^{14}$	$9.4608 \times 10^{16}$
$n \lg n$	62746	2801417	133378058	2755147513	71870856404	86788101610	6861095675070
$n^2$	$10^3$	7746	$6 \times 10^4$	293939	1609969	5615692	56156923
$n^3$	100	391	1532	4421	13737	31594	146646
$2^n$	20	26	32	36	41	45	51
$n!$	9	11	12	13	15	16	17

※1.这里假设每个月都是 30 天，每年都是 365 天。

※2.除了 $f(n) = n!$ ，这里都采用 $f(n) = T$ 把 $n$ 计算出来后采用四舍五入处理。

※3. $f(n) = n!$ 的情况采用 $f(n) \leq T$ 进行计算，然后取 $n_{\max}$ 。