

## 第2章《算法的效率分析基础》习题

1. 指出下列各算法的基本操作：

- (1) 计算  $n$  个数的和。
- (2) 计算  $n!$ 。
- (3) 找出包含  $n$  个数字的列表中的最大值。
- (4) 两个  $n$  位十进制整数相乘。

2. 给定如下插入排序算法：

---

**算法：** InsertionSort //插入排序

---

**输入：**

$A[1:n]$ : 待排序列表

**输出：**

按非降序排序后的列表  $A$

---

**步骤：**

- 1. For  $i=2$  To  $n$  Do
  - 2,      $x \leftarrow A[i]$
  - 3.      $j \leftarrow i-1$
  - 4,     While  $j>0$  And  $A[j]<x$  Do
  - 5.          $A[j+1] \leftarrow A[j]$
  - 6.          $j \leftarrow j-1$
  - 7.     End While
  - 8.      $A[j+1] \leftarrow x$
  - 9. End For
  - 10. Return  $A$
- 

对输入  $\{4, 3, 12, 5, 6, 7, 2, 9\}$ , 算法 InsertionSort 执行了多少次元素比较操作？

3. 按照算法运行时间函数的阶从低到高的顺序，给出下列函数的次序：

$(n-2)!$ ,  $5\log(n+100)$ ,  $2^{n^2}$ ,  $0.001n^4 + 2n^3 + 1$ ,  $\log^2 n$ ,  $\sqrt[3]{n}$ ,  $3^n$

4. 对于如下每一对  $f(n)$  和  $g(n)$ , 要么  $f(n) = O(g(n))$ , 要么  $g(n) = O(f(n))$ , 但不可能两者都成立，确定  $f(n)$  和  $g(n)$  在  $O$  渐进意义下的关系。

(1)  $f(n) = n + 2\sqrt{n}$ ,  $g(n) = n^2$

(2)  $f(n) = n + n\log n$ ,  $g(n) = n\sqrt{n}$

$$(3) f(n) = n + \log n, g(n) = \sqrt{n}$$

$$(4) f(n) = 2(\log n)^2, g(n) = \log n + 1$$

5. 根据  $O$  的定义, 证明  $O(f(n))O(g(n)) = O(f(n)g(n))$ 。

6. 有如下的排序算法:

---

**算法:** Bubblesort

---

**输入:** 待排序列表  $A[1:n]$

**输出:** 按非降序排序后的列表  $A$

---

**步骤:**

```

1.  $i \leftarrow 1$ ;  $Sorted \leftarrow \text{FALSE}$ 
2. While  $i \leq n-1$  And Not  $Sorted$  Do
3.    $Sorted \leftarrow \text{TRUE}$ 
4.   For  $j=1$  Downto  $i+1$  Do
5.     If  $A[j] < A[j-1]$  Then
6.       Swap( $A[j], A[j-1]$ ) //交换  $A[j]$ 和 $[j-1]$ 
7.        $Sorted \leftarrow \text{FALSE}$ 
8.     End If
9.   End For
10.   $i \leftarrow i+1$ 
11. End While
12. turn  $A$ 

```

---

(1) 执行该算法, 元素比较操作最少执行多少次? 什么情形达到该最小值?

(2) 执行该算法, 元素比较操作最多执行多少次? 什么情形达到该最大值?

(3) 执行该算法, 元素赋值操作最少执行多少次? 什么情形达到该最小值?

(4) 执行该算法, 元素赋值操作最多执行多少次? 什么情形达到该最大值?

(5) 使用  $O$  符号和  $\Omega$  符号表示该算法的运行时间。

7. 设计时间复杂度分别为  $O(n)$  和  $\Omega(n \log n)$  算法, 找出包含  $n$  个整数的列表  $A[1:n]$  中的最大值。

8. 有如下的递归算法:

---

$Q(n)$  //输入正整数  $n$

---

**步骤:**

If  $n=1$  Return 1

Else Return  $Q(n-1)+2n-1$

---

(1) 给出该算法执行时乘法运算次数的递推式并求解。

(2) 给出该算法执行时加减运算次数的递推式并求解。

9. 求解以下递推式：

(1)  $T(n)=3T(n-1)-15$ ,  $T(1)=8$

(2)  $T(n)=T(n-1)+n-1$ ,  $T(1)=3$