

算法分析与设计第五周练习

开始时间 2022/03/19 11:10:00

结束时间 2022/03/23 23:59:00

答题时长 6529分钟

答卷类型 标准答案

总分 65

判断题

得分：暂无 总分：5

1-2 用动态规划而非递归的方法去解决问题时，关键是将子问题的计算结果保存起来，使得每个不同的子问题只需要 (5分)
被计算一次。子问题的解可以被保存在数组或哈希散列表中。

☒ T ☐ F

单选题

得分：暂无 总分：10

2-1 在动态规划中，我们要推导出一个子问题的解与其他子问题解的递推关系。要将这种关系转换为自底向上的动态 (5分)
规划算法，我们需要以正确的顺序填写子问题解的表格，使得在解任一子问题时，所有它需要的子问题都已经被
解决了。在下列关系式中，哪一个是不可能被计算的？

☐ A. $A(i, j) = \min(A(i - 1, j), A(i, j - 1), A(i - 1, j - 1))$

☐ B. $A(i, j) = F(A(\min\{i, j\} - 1, \min\{i, j\} - 1), A(\max\{i, j\} - 1, \max\{i, j\} - 1))$

☐ C. $A(i, j) = F(A(i, j - 1), A(i - 1, j - 1), A(i - 1, j + 1))$

☒ D. $A(i, j) = F(A(i - 2, j - 2), A(i + 2, j + 2))$

2-2 给定递推方程 $f_{i,j,k} = f_{i,j+1,k} + \min_{0 \leq l \leq k} \{f_{i-1,j,l} + w_{j,l}\}$ 。要通过循环解此方程，我们一定不能用下列哪种方 (5分)
法填表？

☐ A. `for k in 0 to n: for i in 0 to n: for j in n to 0`

☒ B. `for i in 0 to n: for j in 0 to n: for k in 0 to n`

☐ C. `for i in 0 to n: for j in n to 0: for k in n to 0`

☐ D. `for i in 0 to n: for j in n to 0: for k in 0 to n`

编程题

得分：暂无 总分：50

7-3 矩阵链相乘问题 (20分)

矩阵的乘法定义如下：设A是 $m \times p$ 的矩阵，B是 $p \times n$ 的矩阵，则A与B的乘积为 $m \times n$ 的矩阵，记作 $C = AB$ ，其中，
矩阵C中的第i行第j列元素 c_{ij} 可以表示为： $c_{ij} = \sum_{k=1}^p a_{ik} \times b_{kj} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{ip}b_{pj}$ 。

当多个矩阵相乘时，采用不同的计算顺序所需的乘法次数不相同。例如，A是 50×10 的矩阵，B是 10×20 的矩阵，C
是 20×5 的矩阵，
计算ABC有两种方式： $(AB)C$ 和 $A(BC)$ ，前一种需要15000次乘法计算，后一种则只需3500次。

设 A_1, A_2, \dots, A_n 为矩阵序列， A_i 是阶为 $P_{i-1} \times P_i$ 的矩阵($1 \leq i \leq n$)。试确定矩阵的乘法顺序，使得计算 $A_1A_2\dots A_n$ 过
程中元素相乘的总次数最少。

输入格式:

每个输入文件为一个测试用例，每个测试用例的第一行给出一个正整数 $n(1 \leq n \leq 100)$ ，表示一共有 n 个矩阵 A_1, A_2, \dots, A_n ，第二行给出 $n + 1$ 个整数 $P_0, P_1 \dots P_n$ ，以空格分隔，其中 $1 \leq P_i \leq 100(0 \leq i \leq n)$ ，第 i 个矩阵 A_i 是阶为 $P_{i-1} \times P_i$ 的矩阵。

输出格式:

获得上述矩阵的乘积，所需的最少乘法次数。

输入样例：

在这里给出一组输入。例如：

```
5
30 35 15 5 10 20
```

输出样例：

在这里给出相应的输出。例如：

```
11875
```

7-2 凑零钱 (30分)

韩梅梅喜欢满宇宙到处逛街。现在她逛到了一家火星店里，发现这家店有个特别的规矩：你可以用任何星球的硬币付钱，但是绝不找零，当然也不能欠债。韩梅梅手边有 104 枚来自各个星球的硬币，需要请你帮她盘算一下，是否可能精确凑出要付的款额。

输入格式：

输入第一行给出两个正整数：N (≤ 104) 是硬币的总个数，M (≤ 102) 是韩梅梅要付的款额。第二行给出 N 枚硬币的正整数面值。数字间以空格分隔。

输出格式：

在一行中输出硬币的面值 $V_1 \leq V_2 \leq \dots \leq V_k$ ，满足条件 $V_1 + V_2 + \dots + V_k = M$ 。数字间以 1 个空格分隔，行首尾不得有多余空格。若解不唯一，则输出最小序列。若无解，则输出 **No Solution**。

注：我们说序列 $\{A[1], A[2], \dots\}$ 比 $\{B[1], B[2], \dots\}$ “小”，是指存在 $k \geq 1$ 使得 $A[i] = B[i]$ 对所有 $i < k$ 成立，并且 $A[k] < B[k]$ 。

输入样例 1：

```
8 9
5 9 8 7 2 3 4 1
```

输出样例 1：

```
1 3 5
```

输入样例 2：

```
4 8
7 2 4 3
```

输出样例 2：

```
No Solution
```