

算法分析与设计第十二周练习

开始时间 2022/05/07 22:40:00

结束时间 2022/05/12 23:59:00

答题时长 7279分钟

答卷类型 标准答案

总分 60

判断题

得分：暂无 总分：10

1-1 对N (≥ 2) 个权值均不相同的字符构造哈夫曼树，则树中任一非叶结点的权值一定不小于下一层任一结点的权值。(5分)

☒ T ☐ F

1-2 哈夫曼编码是一种最优的前缀码。对一个给定的字符集及其字符频率，其哈夫曼编码不一定是唯一的，但是每个字符的哈夫曼码的长度一定是唯一的。(5分)

☐ T ☒ F

编程题

得分：暂无 总分：50

7-1 哈夫曼编码 (30分)

本题目要求针对给定的字符串，按照哈夫曼编码原理对其进行编码（即：转换为01串），并输出其对应的哈夫曼编码。注：字符串中的字符按照ASCII码给定序号，如vggba这个字符串中的字符序号分别为43321；构建哈夫曼树时则按照序号顺序排列字符，如权值最小的两个字符为a和b，而不是b和a。

输入格式:

输入一个整数n，然后依次输入n个字符串。

输出格式:

针对每个输入的字符串，输出其哈夫曼编码。

输入样例:

```
2
avvvdddeeeffffggggjk
eeeffrnrjshvhssnn
```

输出样例:

```
11110110110110100100100101101101000000001010101111111110
11011011001001010111111001010000111000110000111111
```

7-2 最短路径算法(Floyd-Warshall) (20分)

在带权有向图G中，求G中的任意一对顶点间的最短路径问题，也是十分常见的一种问题。

解决这个问题的一个方法是执行n次迪杰斯特拉算法，这样就可以求出每一对顶点间的最短路径，执行的时间复杂度为 $O(n^3)$ 。

而另一种算法是由弗洛伊德提出的，时间复杂度同样是 $O(n^3)$ ，但算法的形式简单很多。

在本题中，读入一个有向图的带权邻接矩阵（即数组表示），建立有向图并使用Floyd算法求出每一对顶点间的最短路径长度。

输入格式:

输入的第一行包含1个正整数n，表示图中共有n个顶点。其中n不超过50。

以后的n行中每行有n个用空格隔开的整数。对于第i行的第j个整数，如果大于0，则表示第i个顶点有指向第j个顶点的有向边，且权值为对应的整数值；如果这个整数为0，则表示没有i指向j的有向边。

当i和j相等的时候，保证对应的整数为0。

输出格式:

共有n行，每行有n个整数，表示源点至每一个顶点的最短路径长度。

如果不存在从源点至相应顶点的路径，输出-1。对于某个顶点到其本身的最短路径长度，输出0。

请在每个整数后输出一个空格，并注意行尾输出换行。

输入样例:

```
4
0 3 0 1
0 0 4 0
2 0 0 0
0 0 1 0
```

输出样例:

```
0 3 2 1
6 0 4 7
2 5 0 3
3 6 1 0
```
