**要求代码和实验报告规范，在算法思想中：对实验涉及的数据结构进行有效设计和分析；对算法进行分析并给出时间、空间复杂度的结论；清晰表达实验思路、出现的问题及解决方法。**

一、调试成功程序及说明

1、

题目：图的深度优先和广度优先遍历

算法思想：

1. 算法思想与设计思路

采用邻接矩阵来存放图，DFS采用递归不断进入邻接节点，遍历完就回溯，

BFS采用队列将当前队列中节点的全部邻接节点加入到队列里，一层层遍历

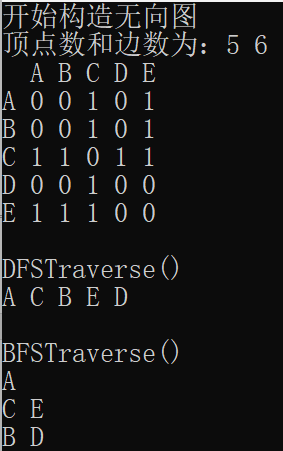
1. 算法复杂度分析

时间复杂度为O(N2),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

运行结果：



结果分析：与预期相同

2、

题目：编程实现Dijkstra算法

算法思想：

1. 算法思想与设计思路

每次从dist里得到未选中的最小节点，然后以此节点为基础，更新起始点到其余未被选中的节点之间的路径距离，然后继续选择，直到所有的顶点被选到。

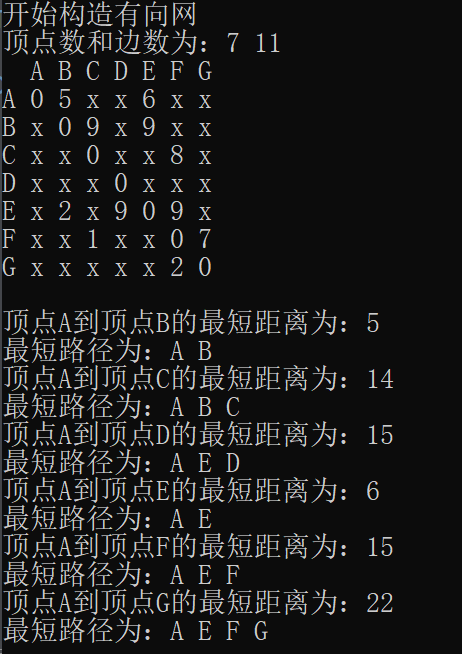
1. 算法复杂度分析

时间复杂度为O(N2),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

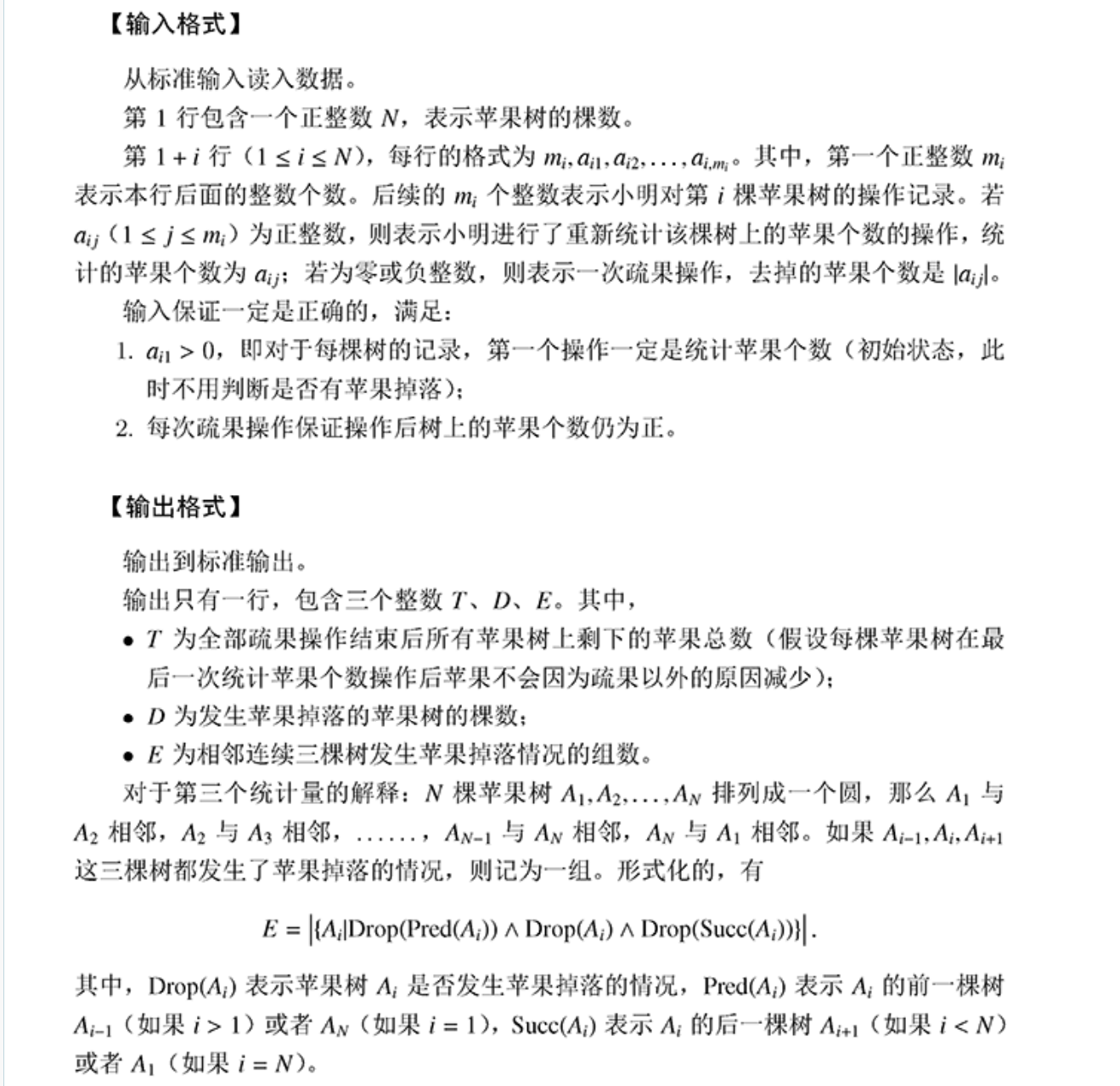
运行结果：



结果分析：与预期相同

3.

题目：



算法思想：

1. 算法思想与设计思路

按照题目要求进行模拟，并记录相应数据

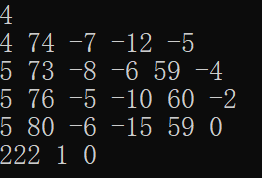
1. 算法复杂度分析

时间复杂度为O(N),空间复杂度为O(1)

1. 实现过程中遇到的问题与解决方法

没有问题

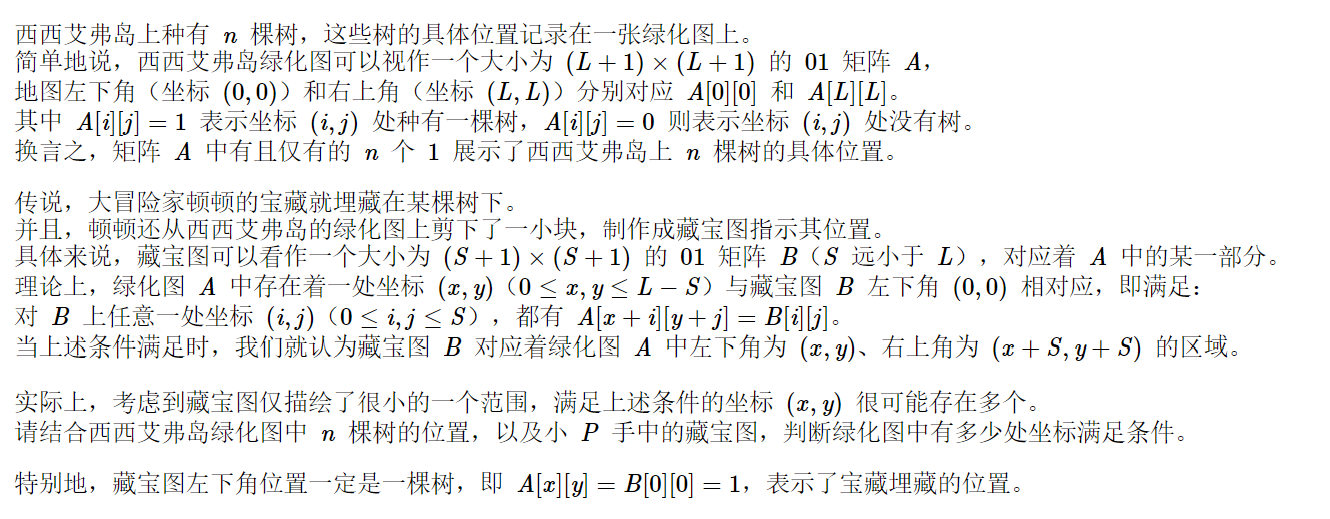
运行结果：



结果分析：与预期相同

4.

题目：



算法思想：

1. 算法思想与设计思路

将输入坐标存入数组中并排序，依次选择合适的坐标作为起始点，并在其后的坐标中选择在藏宝图大小范围内的点插入到局部图中，将局部图与藏宝图比较，若完全符合，则计数加一

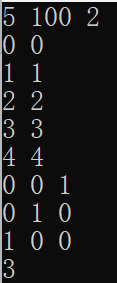
1. 算法复杂度分析

时间复杂度为O(N2),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

运行结果：



结果分析：与预期相同

附加题

T7.27

题目：采用邻接表存储结构，编写一个判别无向图中任意给定的两个顶点之间是否存在一条长度为k的简单路径的算法（一条路径为简单路径指的是其顶点序列中不含有重现的顶点)。

算法思想：

1. 算法思想与设计思路

采用深度优先遍历的方式符合条件的最短路径，一旦找到便立即返回，并用栈存储查找到的最短路径

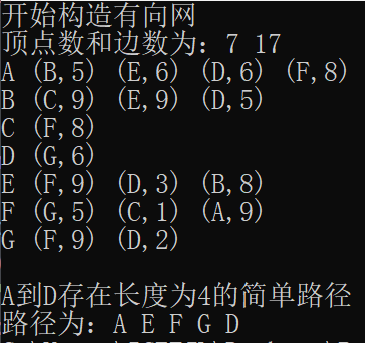
1. 算法复杂度分析

时间复杂度为O(N+E),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

运行结果：

****

结果分析：与预期相同

T7.34

题目：试编写一个算法，给有向无环图G中每个顶点赋以一个整数序号，并满足以下条件：若从顶点i至顶点j有一条弧，则应使i<j

算法思想：

1. 算法思想与设计思路

利用拓扑排序为顶点编号

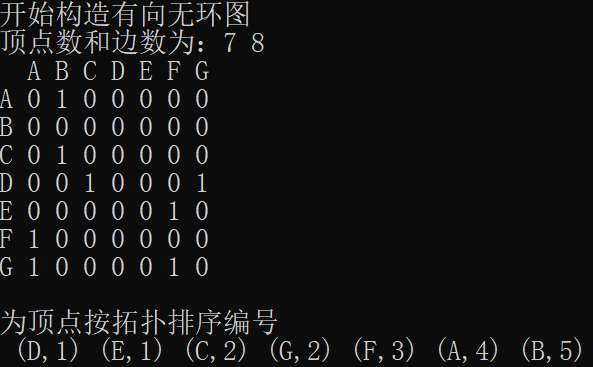
1. 算法复杂度分析

时间复杂度为O(N2),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

运行结果：



结果分析：与预期相同

T7.37

题目：试设计一个求有向无环图中最长路径的算法

算法思想：

1. 算法思想与设计思路

用visited数组存储访问过的节点，用path存储路径，用mlp存储最长路径，遍历每个入度为0的顶点，使用DFS,直到len>maxlen并且到达出度为0的顶点时更新最长路径

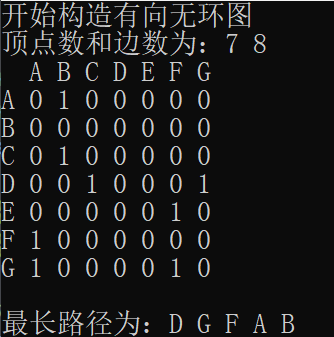
1. 算法复杂度分析

时间复杂度为O(N\*(N+E)),空间复杂度为O(N)

1. 实现过程中遇到的问题与解决方法

没有问题

运行结果：



结果分析：与预期相同

二、代码行数及小结

代码行数为1092行

对于图的各种遍历，找最短路径算法，拓扑排序有了更深的体会，也更熟悉图的各种形式，如无向图，有向图，有向网等