**数据结构课程设计**

**题 目: 模拟导航最短路径**

**班 级：**

**学 生 姓 名：**

**学 生 学 号：**

**2022 年 2月 24日**

目录

[1. 开发环境 3](#_Toc477208932)

[2. 程序的功能 3](#_Toc477208933)

[3. 代码清单 3](#_Toc477208934)

[4. 运行界面 14](#_Toc477208935)

## 开发环境

C++，MSVC 143，VS2022

## 功能描述（内容仅供参考）

给出一张有权有向图和一个起点，求起点到各点的最短路径长度并输出这条最短路径。

输入要求：

首先是n，m，s三个正整数，代表图的顶点数，边数，起点编号

接下来是m行，每一行包含三个数x，y，w，代表从x到y有路径，长度为m

输出格式：

N部分，对于每一个顶点，首先一行输出最短路径长度

接下来一行依次输出这条路径包括的顶点编号，每个编号中间用空格分开

## 代码清单

|  |
| --- |
|  |
| // 作业5控制台.cpp : 此文件包含 "main" 函数。程序执行将在此处开始并结束。  //  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <iostream>  #include <stack>  const int MAXN = 1E3 + 10;  int graph[MAXN][MAXN];  int prev[MAXN], dis[MAXN], visited[MAXN];  int n, m, start;  void doBoom(int start)  {  for (int i = 0; i < n; i++)  {  dis[i] = USHRT\_MAX;  }  dis[start] = 0;  for (int i = 0; i < n; i++)//做N次  {  int u = -1, minv = USHRT\_MAX;  //找当前最短的点  for (int j = 0; j < n; j++)  {  if (visited[j] == 0 && dis[j] < minv)  {  u = j;  minv = dis[j];  }  }  if (u == -1)  {  //提前结束  return;  }  visited[u] = 1;  for (int v = 0; v < n; v++)  {  //松弛所有的边  if (visited[v] == 0 && graph[u][v] != USHRT\_MAX && dis[u] + graph[u][v] < dis[v])  {  dis[v] = dis[u] + graph[u][v];//更新dis  prev[v] = u;//记录最短路径本身（前驱节点）  }  }  }  }  int main()  {  std::cin >> n >> m >> start;  for (int i = 0; i < n; i++)  {  for (int j = 0; j < n; j++)  {  if (i == j)  {  graph[i][j] = 0;  }  else  {  graph[i][j] = USHRT\_MAX;//16位无符号数的最大值，基本上够用了  }  }  }  for (int i = 0; i < m; i++)  {  int x, y, weight;  std::cin >> x >> y >> weight;  graph[x][y] = weight;  }  doBoom(start);//迪杰斯特拉核心部分  for (int i = 0; i < n; i++)  {  std::cout << dis[i] << "\n";//最短路径长度  //接下来输出最短路径经过的点  std::stack<int> stk;  for (int v = i;; v = prev[v])  {  stk.push(v);  if (v == start) { break; }  }  while (!stk.empty())  {  std::cout << stk.top() << " ";  stk.pop();  }  std::cout << "\n";  }  }  // 运行程序: Ctrl + F5 或调试 >“开始执行(不调试)”菜单  // 调试程序: F5 或调试 >“开始调试”菜单  // 入门使用技巧:  // 1. 使用解决方案资源管理器窗口添加/管理文件  // 2. 使用团队资源管理器窗口连接到源代码管理  // 3. 使用输出窗口查看生成输出和其他消息  // 4. 使用错误列表窗口查看错误  // 5. 转到“项目”>“添加新项”以创建新的代码文件，或转到“项目”>“添加现有项”以将现有代码文件添加到项目  // 6. 将来，若要再次打开此项目，请转到“文件”>“打开”>“项目”并选择 .sln 文件 |

## 运行界面（通过图例展示操作流程）

样例输入：

6 8 0

0 1 1

0 3 4

0 4 4

1 3 2

2 5 1

3 2 2

3 4 3

4 5 3

样例输出：

0

0

1

0 1

5

0 1 3 2

3

0 1 3

4

0 4

6

0 1 3 2 5