基于Dijkstra算法的

最短路径规划

课程：网络程序设计

指导老师：张豹

成员：lcc

Sg

时间：2021.06.13

目录

[**一、题目** 2](#_Toc74868076)

[**二、摘要** 2](#_Toc74868077)

[**三、架构设计** 2](#_Toc74868078)

[3.1.流程图 2](#_Toc74868079)

[3.2.Dijkstra.dll 3](#_Toc74868080)

[3.3.Form1.cs[设计] 3](#_Toc74868081)

[3.4.Form1.cs 3](#_Toc74868082)

[**四、关键技术** 4](#_Toc74868083)

[4.1.Dijkstra算法 4](#_Toc74868084)

[4.1.1简介 4](#_Toc74868085)

[4.1.2原理与步骤 4](#_Toc74868086)

[4.2.画图 5](#_Toc74868087)

[**五、结论** 5](#_Toc74868088)

**一、题目**

基于Dijkstra算法的最短路径规划，以武汉大学文理学部、信息学部、工学部为规划区域。

**二、摘要**

本小组项目基于Dijkstra算法，以武汉大学范围（文理学部，工学部，信息学部）为例，设计两种模式—地名输入模式和自由选点模式，并根据输入地名或选择点位自动规划出起点与终点间的最短路径在地图上予以显示，同时显示路径转点和总距离。数据源来自OpenStreetMap,经过Arcmap处理。

**三、架构设计**

本程序为C#语言编写的窗体应用程序(DijkstraForRoutePlanning.exe)，其中包含了一个类库(Dijkstra.dll),若干控件和相应的文件操作以实现目标功能。

## 3.1.流程图

## 3.2.Dijkstra.dll

类库中包含了三个类：structure.cs, read.cs和compute.cs，分别用于存放各种结构体，读取文件和算法计算。其中，

structure.position()用于存放兴趣点，存有地名以及经纬度；

structure.road()存放路径信息；

structure.node()存放节点信息；

read.road()读取路径信息并存储；

read.road2node()将路径信息转换为节点信息；

read.poi()读取兴趣点信息并存储；

compute.shortestPath()用于计算最短路径。

## 3.3.Form1.cs[设计]

程序窗体包含了一个ToolStripMenu,一个DataGridView,四个Lable,四个TextBox,两个Button和一个PictureBox共6类，13个控件并全部实现了功能。其中，

ToolStripMenu用于导入兴趣点文件，如果没有额外的兴趣点文件，将采用默认的兴趣点文件（包含地名和经纬度的文件）。

DataGridView显示兴趣点的名称（各种地标名称），

TextBox分别用于读取或者出发地、目的地和显示路径距离、路径轨迹，

PictureBox用于显示地图和路径，

Button用于选择地名输入模式或自由选点模式。

## 3.4.Form1.cs

应用程序的主体。窗体初始化，调用类库中的类设计了若干方法和事件。

事件 Form1\_Load\_1()：在窗体加载完成时在DataGridView中显示兴趣点地名，供用户在地名输入模式下选择出发地和目的地；

方法InputPoi\_ComputeAndDraw()：地名输入模式路径规划的实现方法；

方法FreeSection\_ComputeAndDrawf()：自由选点模式路径规划的实现方法；

事件 button1\_Click\_1()：执行地名输入模式；

事件 butoon2\_Click\_2()：执行自由选点；

事件 pictureBox1\_MouseClick()：自由选点中标记出发地和目的地；

方法 Near()：寻找兴趣点距离最近的节点；

事件 导入兴趣点文件ToolStripMenuItem\_Click()：导入兴趣点文件。

**四、关键技术**

## 4.1.Dijkstra算法

### 4.1.1简介

Dijkstra算法1959年由E.W.Dijkstra 提出，又称标号法。基本思想：按最短路径长度递增的顺序，逐个产生各最短路径。不仅可以求出起点到终点的最短路径及其长度，而且可以求出起点到其他任何一个顶点的最短路径及其长度

### 4.1.2原理与步骤

1. 用带权的邻接矩阵Cost来表示带权的n个节点的有向图，Cost[i,j]表示弧 <,>的权值，如果从到不连通，则Cost[i,j]=∞。下图表示了一个带权有向图以及其邻接矩阵引进一个辅助向量Dist，每个分量Dist[i]表示从起始点到每个终点的 最短路径长度。假定起始点在有向图中的序号为i0，并设定该向量的初 始值为：

Dist[i]=Cost[,i] ∈V。

令S为已经找到的从起点出发的最短路径的终点的集合。

1. 选择，使得 Dist[j]=Min{ Dist[i]}∈V-S}∈V。就是当前求得的一条从vi0出发的最短路径的终点，令 S=S∪{vj}
2. 修改从出发到集合V-S中任意一顶点的最短路径长度。 如果Dist[j]+Cost[j,k]<Dist[k] 则修改Dist[k]为： Dist[k]=Dist[j]+Cost[j,k]
3. 重复第2、3步操作共n-1次，由此求得从出发的到图上各 个顶点的最短路径是依路径长度递增的序列。

## 4.2.画图

本程序采用PictureBox作为画板，根据数据中的路径信息，采用DrawLine将地图画出，其中还涉及了经纬度转换到像素坐标系中的过程。

**五、结论**

各组件能够正常运行并实现全部功能。程序所规划的最短路径与实际符合的较好，与arcgis中的路径规划结果一致，用户界面整洁美观，简单易用，符合要求。



