

实验：基于最短路径算法的导航软件

问题描述

要求在所给的数据集上建立图结构（邻接矩阵或者邻接表），并在建立的图结构上自行实现Dijkstra算法求解任意两点之间的最短路径。

基本要求

基于 Dijkstra 算法实现一个导航软件，使其可以求解任意输入的源点和目标点之间的最短路径。具体的输入输出形式为：

输入：

- 1、一个txt文档，包含若干行数据，每行数据形式如下：

`src dst distance`

其中src表示源点，dst表示目标点，distance表示从源点到目标点有直接连接的路径且路径长度为Distance

- 2、Src（源点） Dst（目标点） PS：多次输入

输出：

- 1、最短路径长度
- 2、Src到Dst的一条最短路径，例如：Src->p1->p2->p3->...->Dst（逆序输出路径也对）

基本部分的验收方式

实验采用线下检查验收，验收时的测试数据不需要自己准备，这些测试数据包括：

- ◆ 一份 txt 文档

具体检查流程与考察点如下：

- ◆ 运行程序，读入txt文档，并展示所建立的图结构
- ◆ 多次读入源点和目标点，输出最短路径长度和最短路径

(正确实现朴素的dijkstra算法（时间复杂度为 $O(|V|^2)$)并且通过助教验收时给出的测试样例)

基本部分的分数占比为：

- ◆ 代码实现 80%

选做内容

本次实验提供了两个可选的加分项，选做部分的加分占比为

- 使用稀疏图结构的压缩存储方法：5%
- 将dijkstra算法的时间复杂度降低到 $O(|E| * \log(|V|))$ ：10%
- 自由发散部分：鼓励对 Dijkstra 算法做一些额外的探索与尝试，不设严格的限制，你可以根据自己的兴趣和思考去自行选择方向和内容，不论做到什么程度，助教会综合根据思路、难度、工作量来打分。实验鼓励做大胆的、创新的尝试。：5%

注意事项：

- 1、考虑给出的数据集规模比较大的情况
- 2、关于dijkstra算法时间复杂度的降低，有很多种方法，请大家自行搜索资料并学习
- 3、本次实验验收时，并不仅仅是看代码运行的结果，还要检查代码和设计思路，请同学们前来检查实验时整理好思路，能够对着代码讲出：某某代码段是做什么的，为什么要这么做，这么做是如何降低时间复杂度\如何降低存储空间开支（实现压缩存储）的？