

课前签到

数据结构与算法实验课2022秋季



微信扫一扫，使用小程序

1. 微信扫码+**实名**
2. 点击今日签到

签到时间:

16:20~16:35

签到地方:

珠海校区-教学大
楼-A310

人脸识别； 智能定位

实验十五 Dijkstra算法

- **实验目的**
 - **学会利用Dijkstra算法求图的最短路径；**
 - **掌握如何应用Dijkstra算法解决各种实际问题。**



实验十五 Dijkstra算法

- **实验要求**

- **Dijkstra算法的实现以库文件方式实现，不得在测试主程序中直接实现；**
- **程序有较好可读性，各运算和变量的命名直观易懂，符合关键工程要求；**
- **程序有适当的注释。**



实验十五 Dijkstra算法

- 实验任务1:

- 给定 $N*N$ ($2 \leq N \leq 100$) 的方格, 'x'表示障碍, '.'表示没有障碍, 小明可以从一个格子走到他相邻的四个格子, 但是不能走出这些格子。问小明从给定的A点到B点最少需要转90度的弯几次(无法到达则输出-1)

示例:

```
输入规模参数N: 3
输入方格值:
. x A
. . .
B x .
最少转弯次数: 2
```



实验十五 Dijkstra算法

- **实验任务2:**

- **有了一张自驾旅游路线图，你会知道城市间的高速公路长度、以及该公路要收取的过路费。现在需要你写一个程序，帮助前来咨询的游客找一条出发地和目的地之间的最短路径。如果有若干条路径都是最短的，那么需要输出最便宜的一条路径**

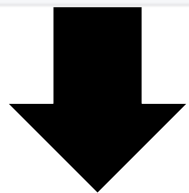


实验十五 Dijkstra算法

● 实验任务2:

示例:

```
4 5 0 3
0 1 1 20
1 3 2 30
0 3 4 10
0 2 2 20
2 3 1 20
```



```
3 40
```

注意:

— 输入: 第1行给出4个正整数 N 、 M 、 S 、 D , 其中 $N(2 \leq N \leq 500)$ 是城市的个数, 假设城市的编号为 $0 \sim (N-1)$; M 是高速公路的条数; S 是出发地的城市编号; D 是目的地的城市编号。随后的 M 行中, 每行给出一条高速公路的信息, 分别是: 城市1、城市2、高速公路长度、收费额, 中间用空格分开, 数字均为整数且不超过500。输入保证解的存在

— 输出: 在一行里输出路径的长度和收费总额, 数字间以空格分隔, 输出结尾不能有多余空格

实验十五 Dijkstra算法

- **实验任务3:**

- **作为一个城市的应急救援队伍的负责人，你有一张特殊的全国地图。在地图上显示有多个分散的城市和一些连接城市的快速道路。每个城市的救援队数量和每一条连接两个城市的快速道路长度都标在地图上。当其他城市有紧急求助电话给你的时候，你的任务是带领你的救援队尽快赶往事发地，同时，一路上召集尽可能多的救援队**

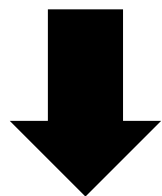


实验十五 Dijkstra算法

● 实验任务3:

示例:

```
4 5 0 3
20 30 40 10
0 1 1
1 3 2
0 3 3
0 2 2
2 3 2
```



```
2 60
0 1 3
```

— 输入：第一行给出4个正整数N、M、S、D，其中 $N(2 \leq N \leq 500)$ 是城市的个数，假设城市的编号为 $0 \sim (N-1)$ ；M是快速道路的条数；S是出发地的城市编号；D是目的地的城市编号；第二行给出N个正整数，其中第i个数是第i个城市的救援队的数目，数字间以空格分隔。随后的M行中，每行给出一条快速道路的信息，分别是：城市1、城市2、快速道路的长度，中间用空格分开，数字均为整数且不超过500。输入保证救援可行且最优解唯一

— 输出：第一行输出最短路径的条数和能够召集的最多的救援队数量。第二行输出从S到D的路径中经过的城市编号。数字间以空格分隔，输出结尾不能有多余空格

实验十五 Dijkstra算法

- 实验任务4:

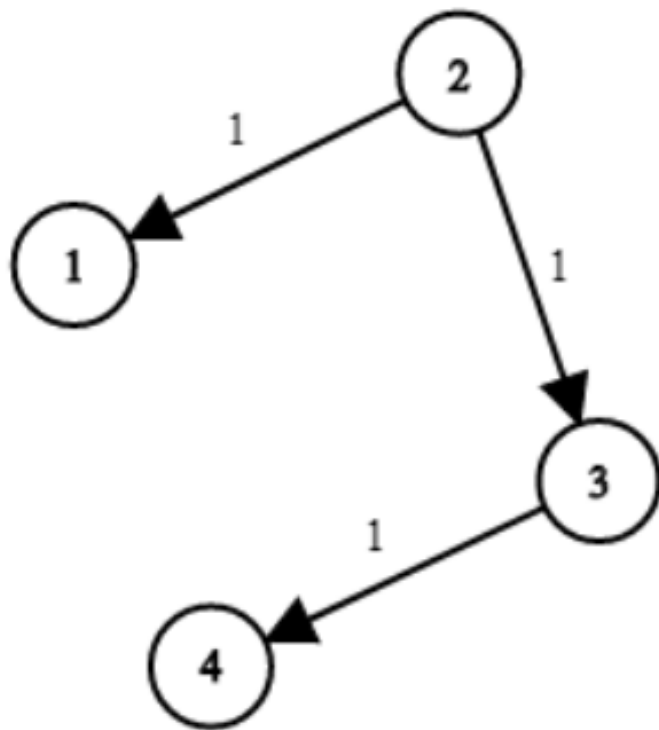
- 有 n 个网络节点，标记为 1 到 n 。给你一个列表 $times$ ，表示信号经过有向边的传递时间。 $times[i] = (u_i, v_i, w_i)$ ，其中 u_i 是源节点， v_i 是目标节点， w_i 是一个信号从源节点传递到目标节点的时间。现在，从某个节点 K 发出一个信号。需要多久才能使所有节点都收到信号？如果不能使所有节点收到信号，返回 -1



实验十五 Dijkstra算法

● 实验任务4:

示例:



注意:

- $1 \leq k \leq n \leq 100$
- $1 \leq \text{times.length} \leq 6000$
- $\text{times}[i].\text{length} == 3$
- $1 \leq u_i, v_i \leq n$
- $u_i \neq v_i$
- $0 \leq w_i \leq 100$
- 所有 (u_i, v_i) 对都互不相同 (即, 不含重复边)

输入: `times = [[2,1,1],[2,3,1],[3,4,1]]`, `n = 4`, `k = 2`
输出: 2