

实习报告

题目：编制一个交通咨询模拟程序，为旅客提供不同要求下的路线规划

组员：吕星宇、胡康、赵昱杰

成稿日期：2022.06.25

一、需求分析

1. 问题描述

处于不同目的的乘客对交通工具不同的要求，比如在旅途中的时间尽可能短，旅费尽可能的省，中转次数最少等。编制一个全国城市间的交通咨询程序，为旅客提供两种或三种最优决策的交通咨询。

2. 功能描述

- **显示列车时刻表**

用户选择功能“显示列车时刻表”，输出现有的列车时刻表，支持翻页的功能。

- **显示飞机航班表**

用户选择功能“显示飞机航班表”，输出现有的飞机航班表，支持翻页的功能。

- **编辑列车时刻表**

用户选择功能“编辑列车时刻表”，可以在列车时刻表中添加或删除车次。

- **编辑飞机航班表**

用户选择功能“编辑飞机航班表”，可以在飞机航班表中添加或删除航班。

- **进行路线规划**

用户输入“起点”“终点”“交通方式”，输出三种决策方式的推荐路线。

3. 测试数据

见文件 TrainTable.txt, FlightTable.txt。

二、设计框架

1. 数据结构

使用邻接表，在表结点中新加一 Info 域存储一趟车次/航班的相关信息，包括 tag 标签（标记这是一趟车次还是航班）、编号、价格、出发时间、到达时间、旅程用时。邻接表示意图如下图所示：

用于进行路线决策，向用户给出三种不同决策方式下的最优路线推荐。

其它文件包括：Flowchart.png，展示程序的执行流程；README.md，对程序整体架构和代码规范做说明；directions.md，用户手册；飞机线路合集.txt 和火车线路合集.txt，含有可批量添加的线路。

三、测试运行结果

① 打开程序即导入已保存线路；

② 功能 1：显示列车时刻表/飞机航班表（以列车时刻表为例）；

```
-----欢迎使用升级版12306-----
                                开发人员：吕星宇、胡康、赵昱杰

如果您还不熟悉本程序，请先查阅说明手册
希望本程序能为您出行提供便利
祝您平安出行，旅途愉快！

请选择功能：(1：显示列车时刻表  2：显示飞机航班表  3：编辑列车时刻表  4：编辑飞机航班表  5：进行路线规划)
1
=====
                        列车车次表
=====
车次   始发站   终到站   出发时间  到达时间  时长(分钟)  票价(元)
T000101 北京     郑州     05:14   14:20     546        163.00
T000102 北京     郑州     06:26   09:10     164        495.00
T000103 北京     郑州     08:00   10:30     150        495.00
T000104 北京     郑州     09:15   11:44     149        495.00
T000105 北京     郑州     09:38   13:01     203        495.00
T000106 北京     郑州     11:48   15:06     198        495.00
T000107 北京     郑州     11:49   17:35     346        251.00
T000108 北京     郑州     14:16   21:23     427        163.00
T000109 北京     郑州     16:00   19:20     200        495.00
T000201 北京     呼和浩特 08:31   11:02     151        326.00
=====
                        第 1 页，共 39 页
=====
请选择.....
(1. 上一页  2. 下一页  3. 跳转到指定页码  4. 退出)
```

③ 功能 2：编辑列车时刻表/飞机航班表（以向列车时刻表中新增、删除车次为例）；

```
请选择您要执行的操作(1/2)
1. 添加车次
2. 删除车次
1
请输入起始站
北京
请输入终点站
烟台
请输入出发时间(时间格式为xx: xx)
12:39
请输入到达时间(时间格式为xx: xx)
16:40
请输入票价
225.50
```

```
请选择您要执行的操作(1/2)
1. 添加车次
2. 删除车次
2
请输入您要删除的车次的编号
T000101
```

④ 功能 3：进行路线决策。

```
请输入您想使用的交通工具(1:火车 2:飞机)
1
请输入您的起点
乌鲁木齐
请输入您的终点
哈尔滨
下面给出推荐的路线
乘坐时间最短的路线:
=====
推荐路线
=====
车次 始发站 终到站 出发时间 到达时间 时长(分钟) 票价(元)
T130804 乌鲁木齐 兰州 19: 45 13: 20 1055 217.00
T080508 兰州 西安 14: 57 18: 07 190 320.50
T050110 西安 郑州 21: 24 04: 52 448 129.00
T010602 郑州 徐州 06: 08 09: 16 188 97.50
T060302 徐州 天津 10: 12 12: 38 146 434.00
T030906 天津 沈阳 14: 34 19: 09 275 321.50
T091609 沈阳 长春 19: 45 21: 17 92 233.50
T162211 长春 哈尔滨 22: 18 23: 15 57 176.50

花费最少的路线:
=====
推荐路线
=====
车次 始发站 终到站 出发时间 到达时间 时长(分钟) 票价(元)
T130801 乌鲁木齐 兰州 10: 31 09: 56 1405 200.00
T080501 兰州 西安 01: 24 10: 01 517 163.00
T050101 西安 郑州 00: 40 07: 05 385 3.00
T010002 郑州 北京 01: 58 08: 16 378 163.00
T000322 北京 天津 21: 39 22: 55 76 23.50
T030901 天津 沈阳 02: 00 09: 05 425 163.00
T091603 沈阳 长春 07: 25 11: 09 224 89.50
T162202 长春 哈尔滨 03: 40 07: 15 215 40.50

中转次数最少的路线:
=====
推荐路线
=====
车次 始发站 终到站 出发时间 到达时间 时长(分钟) 票价(元)
T130801 乌鲁木齐 兰州 10: 31 09: 56 1405 200.00
T080201 兰州 呼和浩特 11: 06 06: 09 1143 236.50
T020001 呼和浩特 北京 13: 20 15: 34 134 328.00
T000301 北京 天津 14: 22 14: 54 32 174.00
T030901 天津 沈阳 02: 00 09: 05 425 163.00
T091601 沈阳 长春 00: 37 04: 16 219 140.50
T162201 长春 哈尔滨 03: 40 07: 15 215 129.50
```

四、调试分析

1. 问题与解决方式

① 空指针 segmentation fault 问题

在增加列车时，针对起始站、终点站均已存在且两地原本没有列车的情况下，实际操作就是在起始站所在的链表的末尾增加一个终点站的节点。为了让一个中间变量指针指向起始站所在链表末尾，将循环设置为：

```
NodeLink *p=CMap->v[depvertex].first;
while(p->vindex!=tervertex && p->next!=NULL)
    p=p->next;
```

但在运行过程中出现了 segmentation fault。分析后发现，若起始站是尚未与任何城市之间有列车，那么它的表头指针指向 NULL，故 p 指向 NULL，那么 p->vindex 便会报错。为

了解决该问题，只需先判断 p 是否指向 NULL：

```
NodeLink *p=CMap->v[depvertex].first;
while(p && p->vindex!=tervertex && p->next!=NULL)
    p=p->next;
```

这样便避免了该问题。

② 最少时间决策权值变化问题

在解决最短时间花费的决策问题中，存在时刻之间的交错，比如北京到郑州的到达时间是 15: 00，要乘坐郑州到乌鲁木齐的车的出发时间是 14: 00，则需要等到第二天才能出发，这样会导致等待时间的增加。

经过讨论，发现问题的根源在于起点的出发时间不确定，所以对这部分算法进行了调整，原先是将起点作为源点，现在将从起点发出的车的每一个线路的终点作为源点，将到达时间作为此源点的出发时刻（这样是固定的），对每个源点均使用 Dijkstra 算法，再选择其中用时最少的。

2. 算法的性能分析

① **时间复杂度**：核心算法利用 Dijkstra 算法，为 $O(n^2)$ 。

② **空间复杂度**：主要存储结构为邻接表，所占空间为 $O(n)$ ；此外还有几个数组辅助结构，所占空间为常数级 $O(1)$ 。

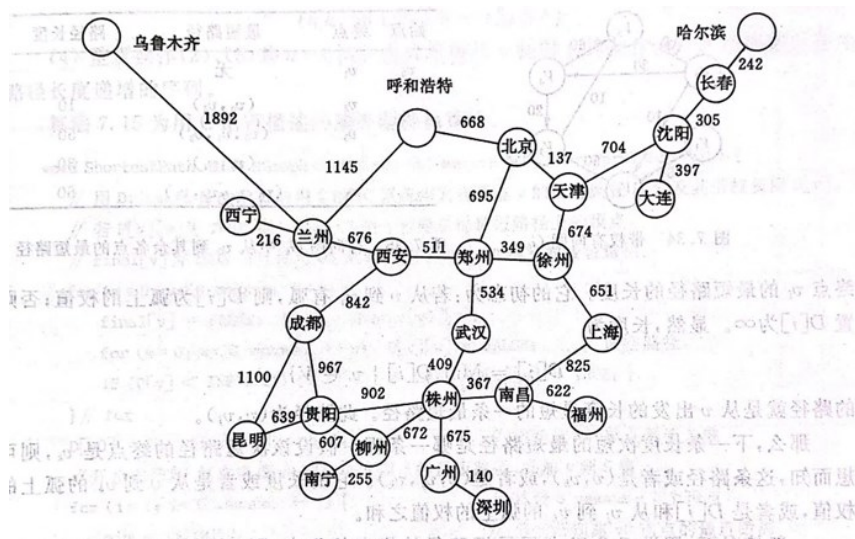
五、用户使用说明

程序开始运行时，会有 1、2、3、4、5 五种选项对应五种功能，用户可以根据自己的需求选择不同的功能。

本程序根据全国交通网络，在 30 座城市中设立了 800 余条火车线路和飞机航班，每一条火车线路与飞机航班都有专属编号，为“XAABBCC”，编号规则为：

- X 为 T 或 F，T 表示火车，F 表示飞机；
- AA 表示始发站城市编号；
- BB 表示终点站城市编号；
- CC 表示相同城市间列车/航班序号。

初始情况下，30 座城市的交通连接情况为：



用户可以选择选项 1 来查看初始的火车线路，也可以选择选项 2 来查看初始的飞机航班。由于列车/航班时刻表过大，用户可通过翻页功能对其进行查看。

在初始 30 座城市和 800 余条线路的基础上，用户可以按照个人意愿增加城市/线路。用户可以选择选项 3 来修改火车线路，也可以选择选项 4 来修改飞机航班。

无论是修改火车线路还是飞机航班，都可以选择增加或删除两种功能：

- 若要**增加线路**，用户需要按照提示依次输入始发站、终点站、起始时间、终止时间、票价。程序会自动生成线路编号并保存至线路表；
- 若要**删除线路**，用户只需输入编号，系统便会完成删除。

本程序的核心功能是进行路线规划。用户可以选择选项 5 来规划最便利、最实惠的路线。根据出行工具，用户可以选择火车或飞机，然后输入始发地和目的地。程序会规划出乘坐时间最短、花费最少、中转次数最少三种最优方案。

六、附录

您可在 <https://github.com/Zemdalk/DS-project-work> 的子文件夹“NationalTrafficConsult”下查看全部代码和相关文件。