

哈尔滨工业大学（深圳）

# 数据结构实验报告

---

图形结构及其应用

学 院： 计算机科学与技术  
姓 名： 胡博涵  
学 号： SZ170110113  
专 业： 计算机类  
日 期： 2018.6.3

## 一、问题分析

### 任务一：

任务一的具体场景是给出一个输入序列，对应每一门课程的先修课程要求，对于这个序列，给出符合要求的课程学习顺序。这实际上是有向无环图的拓扑排序。对于计算机而言，要用特定的数据结构存储有向无环图，并在该有向无环图中，以特定算法求出其中的一个拓扑序列。因此，问题就转换成：将特定的输入序列转换成图结构进行存储，并对图进行拓扑排序。

### 任务二：

任务二的具体场景是给出形如  $a/b = 2$ ， $b/c = 3$  等若干个除法式子，已知他们的结果。之后给出几个形如  $a/c$  的未知的除法式子，通过已知的这几个式子求出未知量。由于给出的式子均为二元表达式，且具有顺序性、传递性，由图的定义，可以将除法关系表达为顶点之间的有向边。实际上，如果将除数和被除数看作图中的顶点，那么除法关系表示为从被除数a到除数b有一条有向边，其权值即为a/b的结果。这样，就可以把除法关系抽象为有向图。

### 任务三：

任务三的具体场景是公交线路查询系统的实现，由于涉及到公交线路和公交站点，对于计算机而言，适合采用图的结构进行存储。对于公交线路查询、换乘线路的设计，也涉及到图的基本算法：深度优先搜索、最短路径算法。

## 二、设计思想

### 2.1 详细设计

#### 任务一：

任务一给出的图结构实际上是一个AOV网，要安排事件的先后顺序，需要进行拓扑排序。即不断地在所有的顶点中，选出入度为0的顶点，删除这个顶点，将该顶点放入队列中，并更新其余顶点的入度，直到图中所有的顶点度数都为0，就完成了拓扑排序。若图中不存在入度为0的顶点，则证明图中有环，拓扑排序失败。本题采用的存储结构为邻接矩阵。

#### 任务二：

任务二中，将除法等式的二元关系转换为了有向带权图的结构，用从被除数a指向除数b，权值为结果c的有向边代表一个除法式子。

这样，就可以把求未知表达式结果的问题转换为在有向图中求某条路径的问题，只要条件充足，给出一个结果未知的等式a/d，就可以求出对应式子的值。

其主要思路为在图中，寻找从顶点a到d到一条路径，计算结果就是路径上的每一个边的权值之积。

对于路径的寻找算法，可以采用Dijkstra算法或Floyd算法寻找最短路径，也可以采用试探回溯法寻找任意路径。用一个队列记录这个路径上的顶点，以便于计算最终的结果。本题采用的存储结构为邻接矩阵。

#### 任务三：

任务三要完成一个系统的设计。分为存储部分和查询部分。其中存储部分使用邻接矩阵和关联矩阵相结合的方式。

为方便存储对应，对站点的存储使用一维结构体数组的方式，每一个数组单元编号唯一地对应一个站点名称。

每一个站点的结构体存储着其站点名、经过站点的公交线路和在每一条线路的相邻顶点（方便删除顶点）。

在邻接矩阵中，每一个顶点代表一个公交站，顶点之间的无向连边代表两个站点位于同一条公交线路的相邻的站点，连边中存储的信息也包括两相邻顶点间的所有公交线路编号。

在关系矩阵中，每一个顶点都保存有与之相关联的公交线路信息。在查询换乘线路时，可以采用Dijkstra算法或Floyd算法寻找最短路径，也可以采用DFS搜索所有路径并找出最短的一条。

添加线路时，首先读取线路的编号，再以“-”分割线路字符串，将得到的字符串存入一个数组temp，在顶点向量中定位字符串对应的编号（若不存在，则添加），将当前线路存入对应编号的邻接矩阵中，同时记录每一个顶点在当前线路的前驱和后继。  
在查询两个顶点之间是否有直达线路时，只需查询关联矩阵中，两个顶点是否与同一条边相关联即可。

## 2.2 存储结构及操作

### 任务一：

存储结构：

数据类型	功能	备注
二维整型数组AdjMatrix	存储邻接矩阵数据	为1，代表有边相连
一维整形数组Stack	模拟栈	\
一维数组inDegree	记录顶点入度	\

相关操作：

- \* 解析输入序列：第一个数字为顶点个数，后面一律跳过无关字符，每两个数字为一组，建立有向边。
- \* 查找入度为0的顶点：线性扫描入度序列InDegree，找出值为0的位置。
- \* 删除编号为i的顶点：线性扫描邻接矩阵AdjMatrix中的第i行，将标记为1的位置A[i][j]改为0，并在入度列inDegree中把对应位置inDegree[j]递减1

### 任务二：

存储结构：

数据类型	功能	备注
二维整型数组AdjMatrix	存储邻接矩阵数据	AdjMatrix[i][j]=i号顶点/j号顶点的值
二维整形数组isVisited	记录图中的顶点是否被访问过	为1时代表被访问过
一维整形数组Stack	模拟栈进行递归调用	\
二维字符数组vertex	记录顶点的名称信息	将每一个未知数与唯一编号一一对应

相关操作：

函数名	功能	参数	返回值
locate_Vex	定位对应名称顶点在数组中的位置	字符串：顶点名称	顶点编号(不存在返回-1)
str2exp	解析表达式，分离出被除数、除数和商	字符串：表达式	表达式结构体Expression
addEdge	向无向图中加入边（添加除法关系）	Expression结构体	无
nextNbr	寻找当前顶点的下一个邻居	整数：顶点编号	整数：顶点编号
calc	计算某个表达式的值	被除数、除数顶点编号	计算结果(无法计算返回-1)

### 任务三：

存储结构：

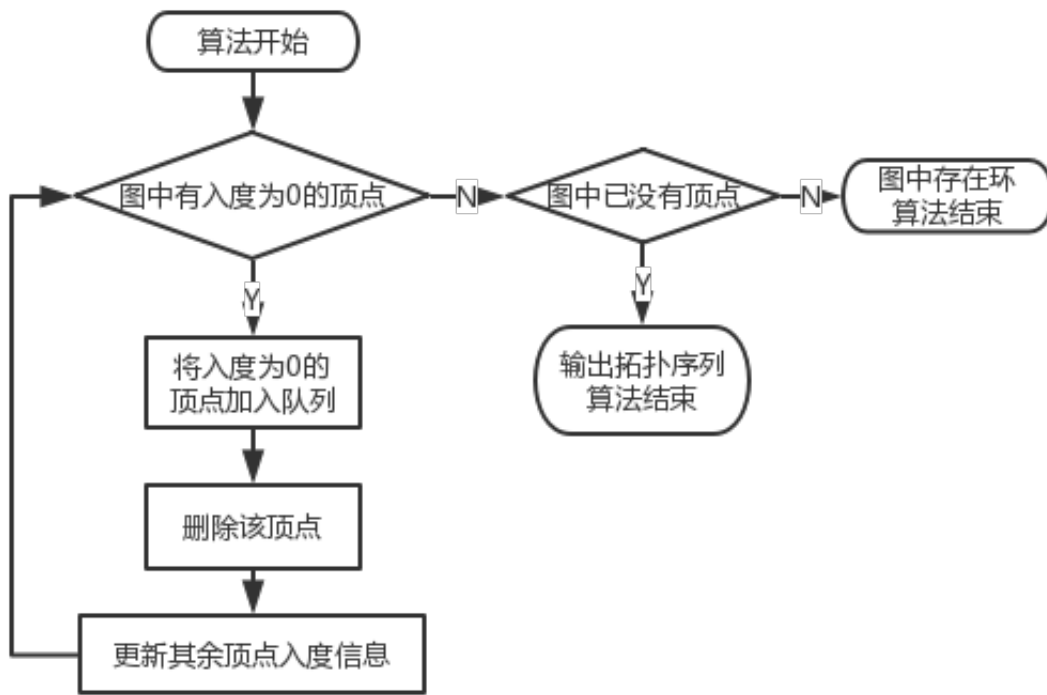
数据类型	功能	备注
结构体类型Arc	存储两个站点之间的线路数据	成员变量：整形数组（存储公交车线路号）、整形变量（存储公交线路数）
结构体类型Vertex	存储公交站点信息	成员变量：字符串（公交站名）、整形数组（存储公交车线路号）、整形变量（存储公交线路数）、二维数组（存储该公交站点在每一条线路的前驱和后继）
结构体类型Graph	存储公交线路信息	成员变量：结构体Vertex数组（将每一个站点与唯一编号一一对应）、整型变量（存储站点数量）、二维结构体Arc数组AdjMatrix（存储邻接矩阵公交线路信息）
结构体类型Route	在Dijkstra算法中，存储最短路径信息	成员变量：字符数组（存储源点到该点路径走法）、整型变量（存储源点到该点最短距离）

相关操作：

函数名	功能	参数	返回值
locate_S tation	定位对应名称车站在数组中的编号	字符串：顶点名称	顶点编号(不存在返回-1)
clear_m ap	清空邻接矩阵的所有信息	结构体Graph指针	无
addLine	向无向图中加入公交线路	结构体Graph指针、公交线路序列 字符数组	无
short_P ath	用Dijkstra算法求解图中最短路	字符串：出发点名称、目的地名称， 结构体Graph指针	结构题Route指针（若不存在车站，返回NULL）
del_Stat ion	删除某个站点	字符串：待删除的站点名、结构体G raph指针	0:删除成功 -1:删除失败

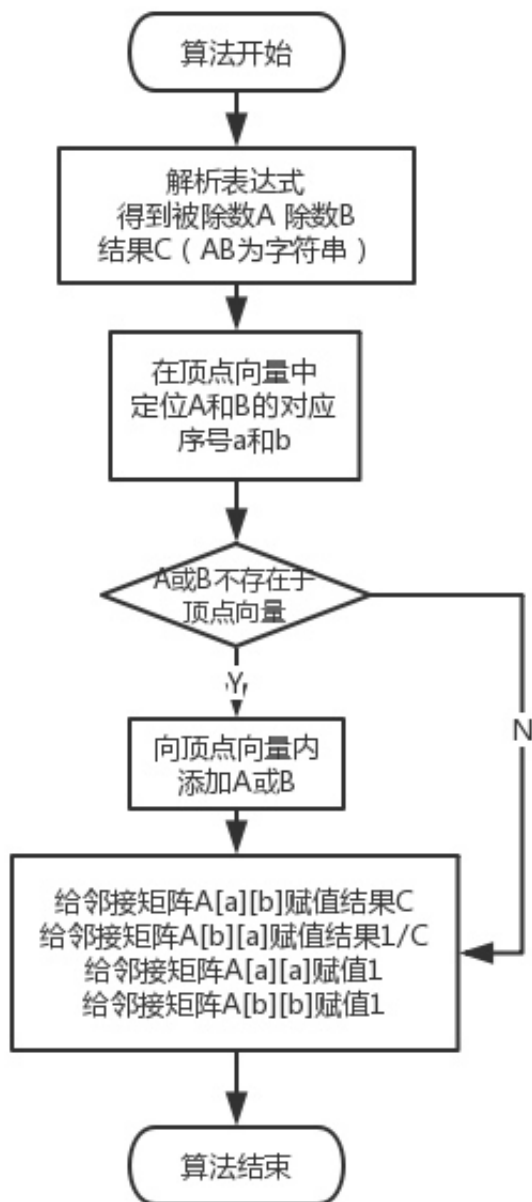
2.3 程序运行整体流程

任务一：

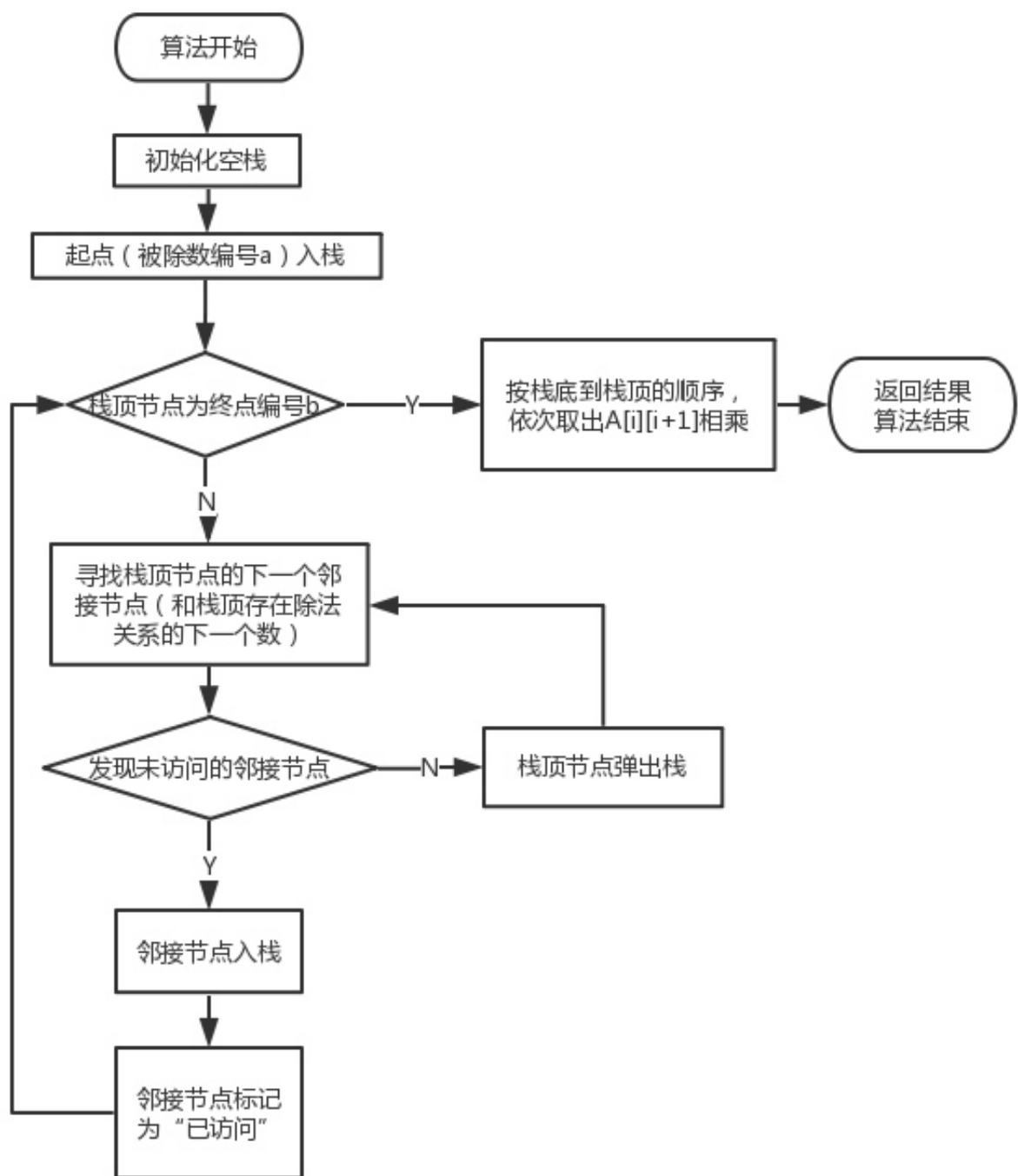


任务二：

记录已知方程式：

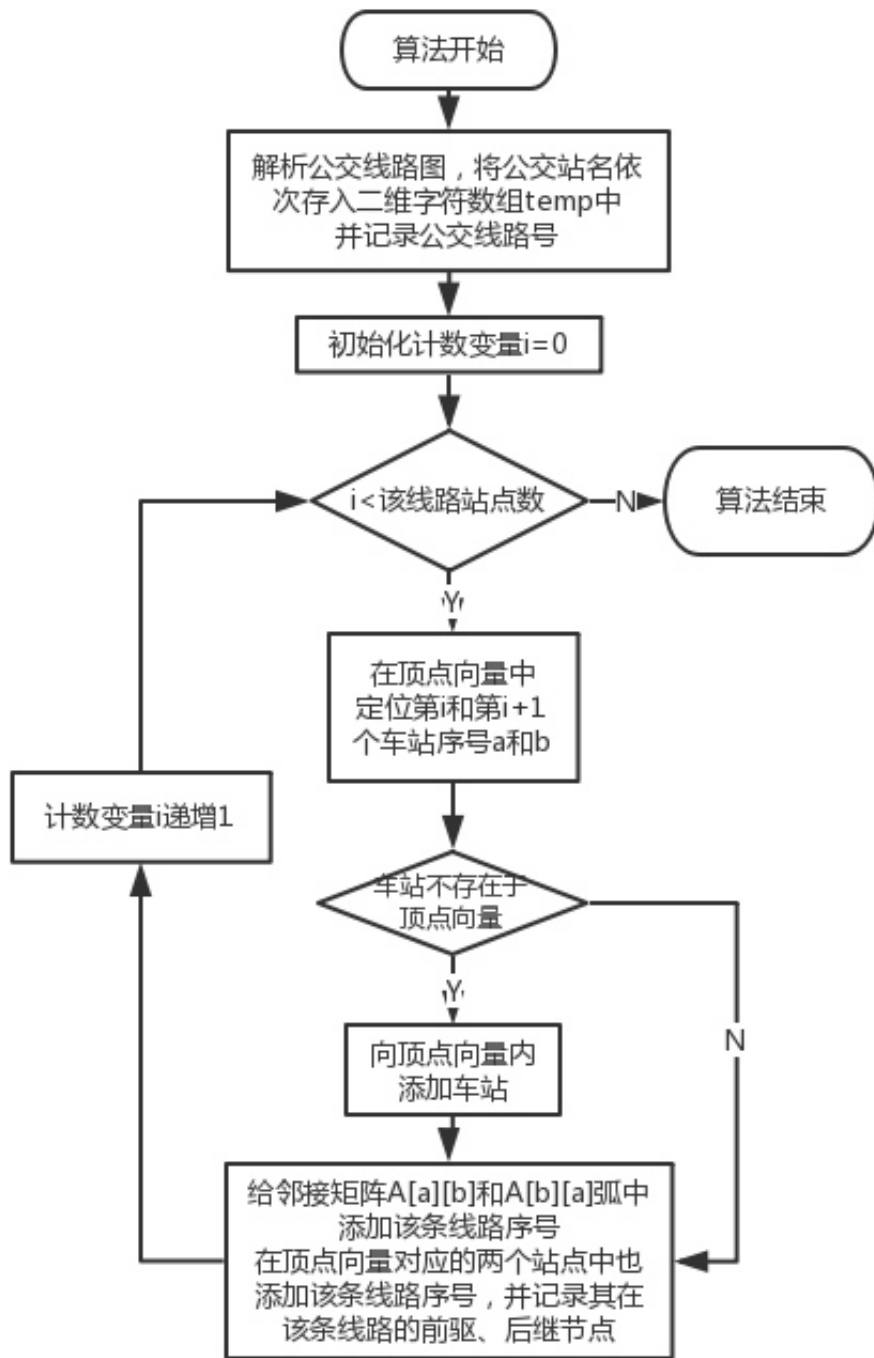


计算未知方程式（类似于迷宫路径算法）：



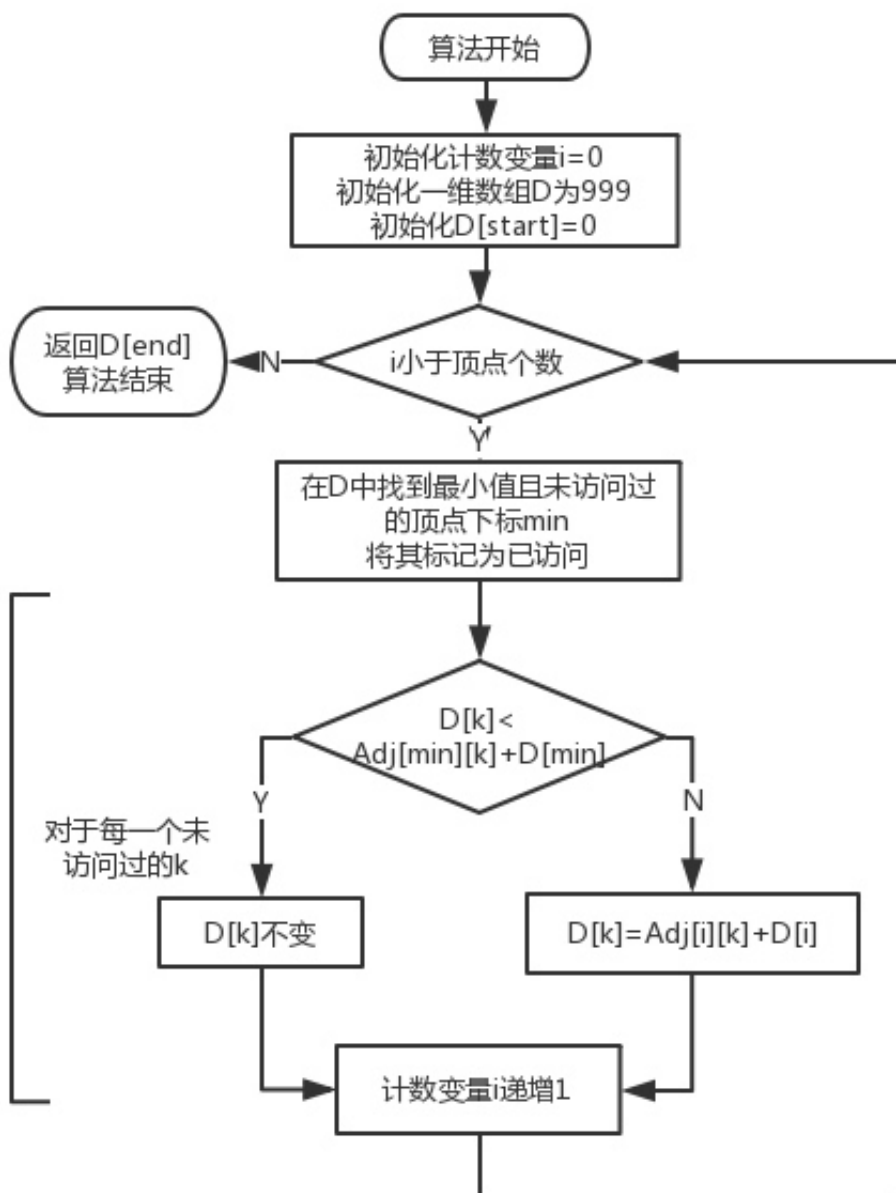
任务三：

存储公交线路

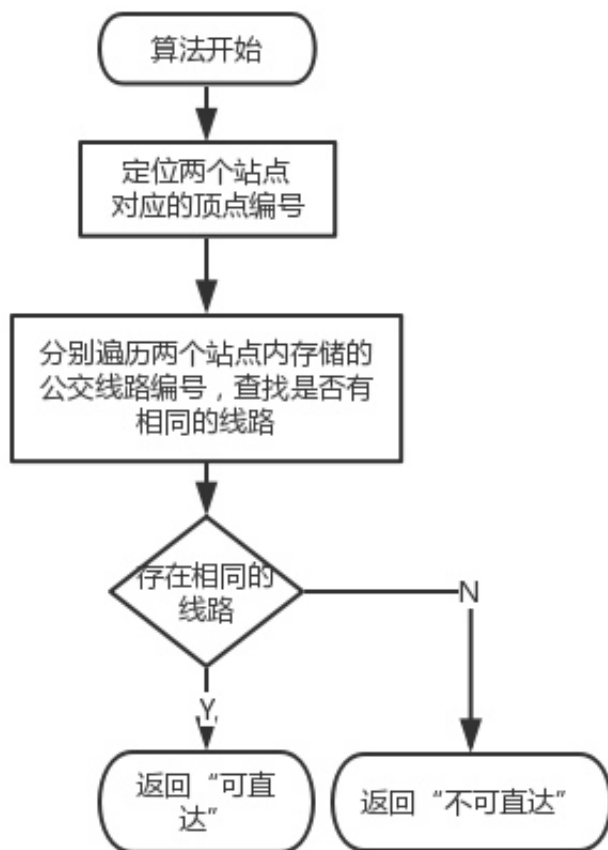


### 查找最短路径（Dijkstra算法）

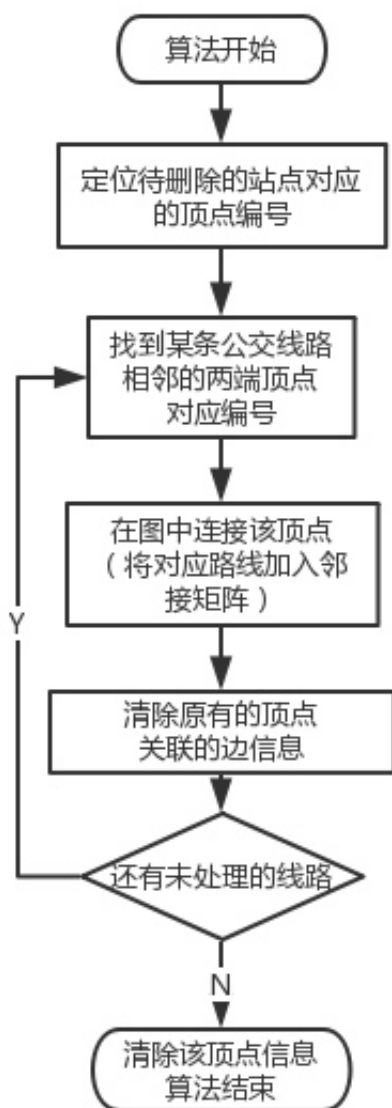




查询两个站点之间是否直达



删除车站



### 三、用户手册

#### 任务一：

输入数据的方式：输入一串序列例如  $4, [[1,0], [2,0], [3,1], [3,2]]$ ，第一个数字是顶点的个数，接下来用方括号括起来几组二元关系例如  $[a,b]$ ，表示  $b$  为完成  $a$  的先决条件。输入数据的方式如上所示。

输出数据的方式：输出完成这些事件的顺序。若输入不合法（存在环等情况），则返回错误信息。

#### 任务二：

输入数据的方式：先输入几个以逗号分隔的除法方程组，例如  $a/b=2, c/d=2, a/c=4$ ，再输入待求的几个式子，以逗号分隔，例如  $c/a, b/c$ 。

输出数据的方式：输出计算的结果，以逗号分隔，在上例中，为  $0.25, 2$ 。

#### 任务三：

程序运行后，会显示一个菜单：

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

输入数据的方式：文件打开时，会自动读取工作目录下的公交站点信息（保存在data.txt中），格式如下：

开头为线路号，后面紧跟站点信息，相邻站点用‘-’分隔

- 1 民治民塘路公交总站-铁路桥-彩悦大厦-长岭陂地铁站-南方科技大学-丽水路南-哈工大园区-桃源康复中心-红花岭-珠光北路-桃源文体中心-珠光村-宝珠花园-军休二所-新车管所-白石洲-世界之窗地铁接驳站
- 2 西乡大铲湾-宝安人民医院-兴东地铁站-深职院-西丽医院-西丽劳力市场-红花岭-平山村南-长岭陂地铁站-深圳北汽车站-五和大道-坂田地铁站-平安大道总站
- 3 民治民塘路公交总站-铁路桥-民乐地铁站-北大医院-桃源村-新屋村-茶光-西丽法庭-九祥岭-深职院-官龙村-南岗工业区-石岩综合场站
- 4 民田路公交总站-购物公园-会展中心-市民中心-深圳北汽车站-长岭陂地铁站-塘朗小学-塘朗村口-塘朗派出所-红花岭-西丽劳力市场-王京坑村
- 5 王京坑村-大袋小学-西丽湖-动物园-深职院-西丽法庭-茶光-白石洲-世界之窗地铁接驳站
- 6 深圳北站-长岭陂地铁站-平山村南-红花岭-西丽劳力市场-西丽法庭-西丽社区-茶光-朗山路中-朗山路口-科苑北环立交-科苑立交南-科苑北-深南科苑立交北-高科技中心-科苑地铁站-深圳湾口岸
- 7 南山中心区-后海地铁站-科苑地铁站-深港产学研-高科技中心-科苑北-科苑立交南-科苑北环立交-华翰科技-茶光-西丽社区-西丽医院-留仙洞地铁站-深职院-中兴人才公寓
- 8 马上头总站-深职院-西丽法庭-茶光-科苑北环立交-科苑立交南-科苑北-高科技中心-科苑南路口-深圳湾口岸

### 添加站点信息

程序运行时，输入1，回车读取文件中多条线路信息。

也可以输入2，回车后，手动输入单条线路，格式同上。

### 删除站点信息

输入3，回车后，输入待删除的站点名，若找到站点成功删除，则输出“删除成功”，否则输出“未找到站点，删除失败”。

### 查询最短路径

输入4，回车后，依次输入两个站点名称，若两个站点间具有最短路径，将输出最短路径，若两个站点间无法到达，则输出“无公交线路连通”。

### 查询是否直达

输入5，回车后，依次输入两个站点名称，若两个站点间能直达，则会输出直达的线路，同时输出某种换乘方案，若不能直达，则会输出路径最短的换乘线路。

## 四、总结

本次实验主要涉及到了图的存储结构。

任务一的课程表问题，实际上就是有AOV网的事件时序安排问题，涉及到的主要算法为拓扑排序。

任务二的除法问题，经过分析可知，其结构特别类似于图的有关结构。给出的每一个方程均可以对应图中的一条边，两个顶点分别为除数和被除数，权重为该算式的结果。由于除法计算可以分解分步计算的特性，就可以将求

解未知式子结果的问题转换为分步求解若干个已知式子结果的问题，其对应的图论问题就是在图中寻找被除数到除数的一条路径，路径的权值相乘即为算式的结果，寻径问题涉及的主要算法为试探与回溯，使用栈的结构模拟递归实现。

任务三中，需要对公交线路进行存储，考虑到存储复杂性和时间效率问题，采用邻接矩阵的方式进行存储，其中邻接矩阵的每一个单元存储的弧的信息为经过该边的所有线路。为方便直达的查询，同时采取类似关系矩阵的储存方法，对每一个顶点存储与其关联的线路，在查询直达时，就可以通过比较经过两个站点的线路是否有交集，从而得出是否存在直达线路。对于最短路径的求解和换乘路径的查询，采用Dijkstra算法来实现。

遇到的问题：由于结构体内包含过大的数组，导致程序运行时栈溢出错误（0xC00000FD），必须将变量定义在堆上（全局变量）才能解决。同时，由于函数调用的时候，传递的参数是副本，所以若将一个过大的变量传入函数，也可能造成栈溢出，传入指针可以解决这个问题。

## 五、结果

### 程序运行正确结果截图

任务一：

 C:\Users\hubohan\Desktop\Lab4\_T1\bin\Debug\Lab4\_T1.exe

请输入序列：

7, [[1, 0], [2, 0], [3, 1], [3, 2], [4, 3], [5, 4], [6, 4]]  
0 1 2 3 4 5 6

Process returned 0 (0x0)      execution time : 0.032 s  
Press any key to continue.

 C:\Users\hubohan\Desktop\Lab4\_T1\bin\Debug\Lab4\_T1.exe

4, [[1, 0], [2, 0], [3, 1], [3, 2]]  
0 1 2 3

Process returned 0 (0x0)      execution time : 1.402 s  
Press any key to continue.

 C:\Users\hubohan\Desktop\Lab4\_T1\bin\Debug\Lab4\_T1.exe


请输入序列:

3, [[0, 1], [1, 2], [2, 0]]

图中存在环, 无法拓扑排序。

Process returned 0 (0x0)      execution time : 0.027 s  
Press any key to continue.

任务二:

 C:\Users\hubohan\Desktop\Lab4\_T2\bin\Debug\Lab4\_T2.exe

请输入已知的表达式:  $a/b=0.4$ ,  $b/d=2$ ,  $c/a=4$

请输入待计算的表达式:  $a/d$ ,  $b/c$

0.8, 0.625

Process returned 0 (0x0)      execution time : 0.099 s  
Press any key to continue.

 C:\Users\hubohan\Desktop\Lab4\_T2\bin\Debug\Lab4\_T2.exe

请输入已知的表达式:  $aa/bb=2.0$ ,  $bb/cc=2.0$

请输入待计算的表达式:  $aa/cc$ ,  $cc/aa$

4, 0.25

Process returned 0 (0x0)      execution time : 26.329 s  
Press any key to continue.



任务三：

手动添加公交线路

```
C:\Users\hubohan\Desktop\Lab4_T3\bin\Debug\Lab4_T3.exe
从文件读取数据成功！
1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出
请输入你的选择：1
请输入路线：8马上头总站-深职院-西丽法庭-茶光-科苑北环立交-科苑立交南-科苑北-高科技中心-科苑南路口-深圳湾口岸
添加成功！
1. 手动录入单条公交信息
```

删除公交站点

```
C:\Users\hubohan\Desktop\Lab4_T3\bin\Debug\Lab4_T3.exe
从文件读取数据成功！
1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出
请输入你的选择：2
请输入要删除的站点：马上头总站
删除成功！
1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出
请输入你的选择：2
请输入要删除的站点：哈工大
删除失败，未找到站点！
```

从文件读取数据成功！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：3

请输入起点站：平山村南

请输入终点站：朗山路中

最短路径：5

其中的某条最短路径为：平山村南--6/2-->红花岭--6/4/2-->西丽劳动市场--6-->西丽法庭--8/5/3-->茶光--6-->朗山路中

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：2

请输入要删除的站点：西丽法庭

删除成功！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：3

请输入起点站：平山村南

请输入终点站：朗山路中

最短路径：5

其中的某条最短路径为：平山村南--6/2-->红花岭--6/4/2-->西丽劳动市场--6-->西丽社区--7/6-->茶光--6-->朗山路中

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：3

请输入起点站：西丽法庭

请输入终点站：平山村南

未找到相关车站！

最短路径查询：

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：3

请输入起点站：哈工大园区

请输入终点站：深圳湾口岸

最短路径：11

其中的某条最短路径为：哈工大园区--1-->桃源康复中心--1-->红花岭--6/4/2-->西丽劳动市场--6-->西丽法庭--8/5/3-->茶光--8-->科苑北环立交--8/7/6-->科苑立交南--8/7/6-->科苑北--8/7-->高科技中心--6-->科苑地铁站--6-->深圳湾口岸



C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择: 3

请输入起点站: 哈工大威海

请输入终点站: 深圳湾口岸

未找到相关车站!

直达路径查询: (同时, 若有换乘方案, 也会给出)

C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

从文件读取数据成功!

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择: 4

请输入起点站: 哈工大园区

请输入终点站: 南方科技大学

从 哈工大园区 到 南方科技大学 存在直达路径1路。

C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择: 4

请输入起点站: 马上头总站

请输入终点站: 深圳湾口岸

从 马上头总站 到 深圳湾口岸 存在直达路径8路。

C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择: 4

请输入起点站: 科苑北环立交

请输入终点站: 深圳北站

从 科苑北环立交 到 深圳北站 存在直达路径6路。

从文件读取数据成功！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：4

请输入起点站：哈工大园区

请输入终点站：科苑北环立交

从 哈工大园区 到 科苑北环立交 不存在直达路径，可按以下路径换乘：

哈工大园区--1-->桃源康复中心--1-->红花岭--6/4/2-->西丽劳动市场--6-->西丽法庭--8/5/3-->茶光--8-->科苑北环立交

最短路径：6

找不到车站：

C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

从文件读取数据成功！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：4

请输入起点站：哈工大园区

请输入终点站：西丽

未找到相关车站！

异常处理：

C:\Users\hubohan\Desktop\Lab4\_T3\bin\Debug\Lab4\_T3.exe

文件data.txt不存在，请手动录入！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：\_

从文件读取数据成功！

1. 手动录入单条公交信息
2. 删除公交站点
3. 查询最短路径
4. 查询是否直达
0. 退出

请输入你的选择：sf sd

非法输入，请重新输入！

请输入你的选择：dfs

非法输入，请重新输入！

请输入你的选择：哈工大

非法输入，请重新输入！

请输入你的选择：