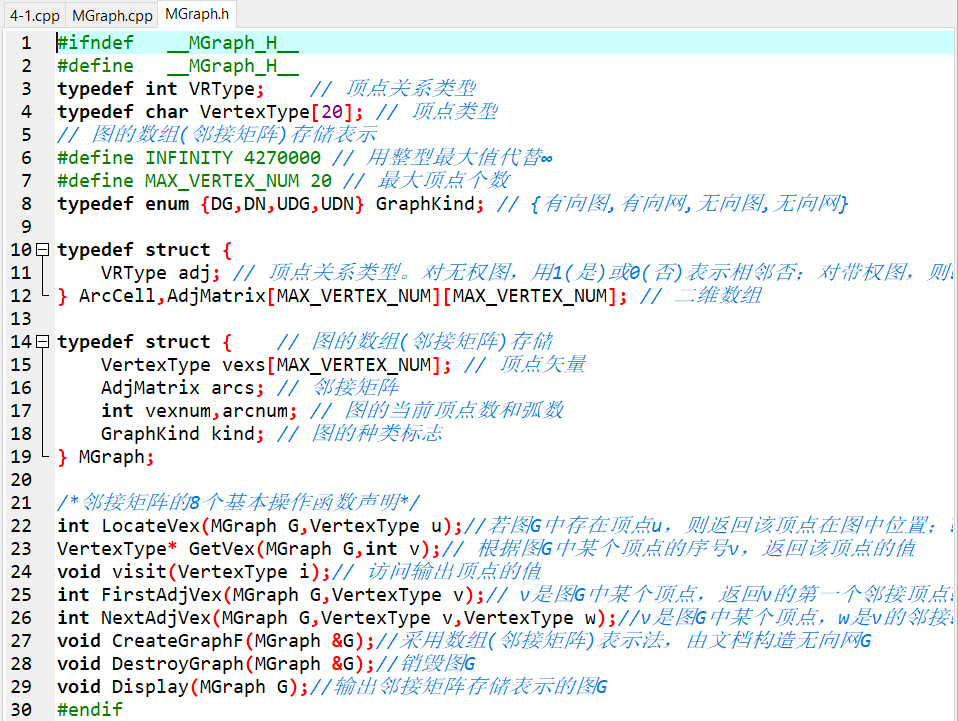
**数据结构与算法 第四次实验**

学号：22920212204392 姓名：黄勖

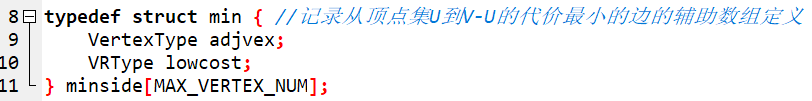
1. **实验目的**
2. 了解分别使用邻接矩阵与邻接表存储图的基础实现方法与原理，理解存储图的基本操作的代码编写方式
3. 了解分别使用Prim算法Kruskal算法的实现方法与原理，在理解的基础上学习代码编写
4. 学会灵活按照邻接矩阵与邻接表的存储方式自由编写存储结构代码
5. 通过实验探索图的深度优先搜索方法，理解回溯的意义
6. **实验内容**

4-1 假设带权连通图G有n个顶点，用邻接矩阵A[n][n]表示存储结构，u为指定顶点的序号。试设计Prim算法，用于从顶点u出发构造连通图G的最小生成树。

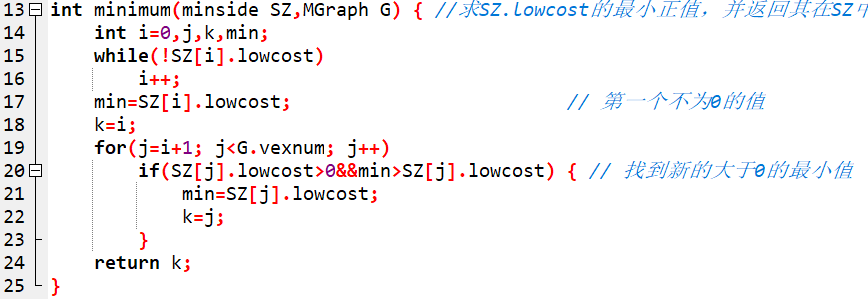
* 图的基本操作包括在MGraph.h中



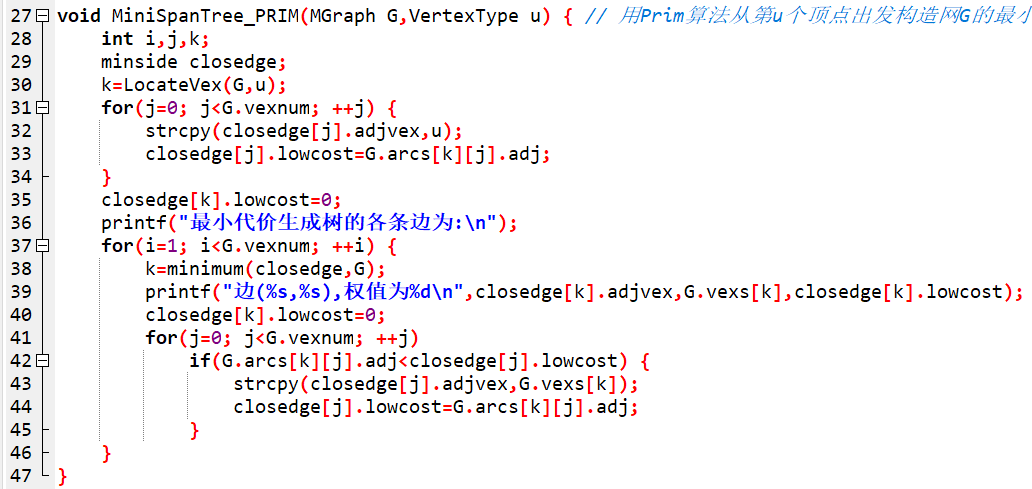
* 记录从顶点集U到V-U的代价最小的边的辅助数组定义



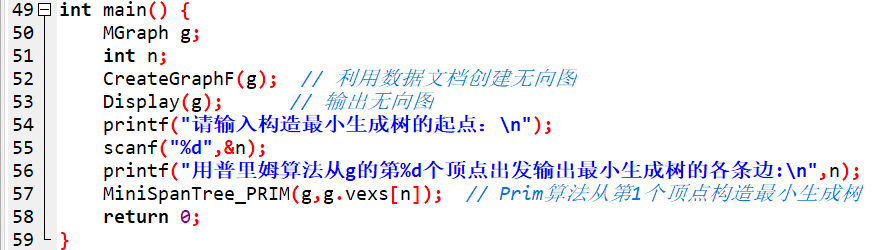
* 求SZ.lowcost的最小正值，并返回其在SZ中的序号



* 用Prim算法从第u个顶点出发构造网G的最小生成树T，输出T的各条边



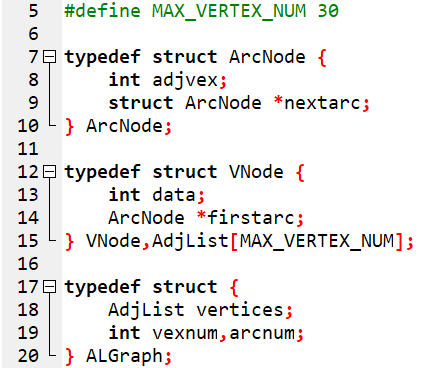
* Main函数创建无向图、构造最小生成树



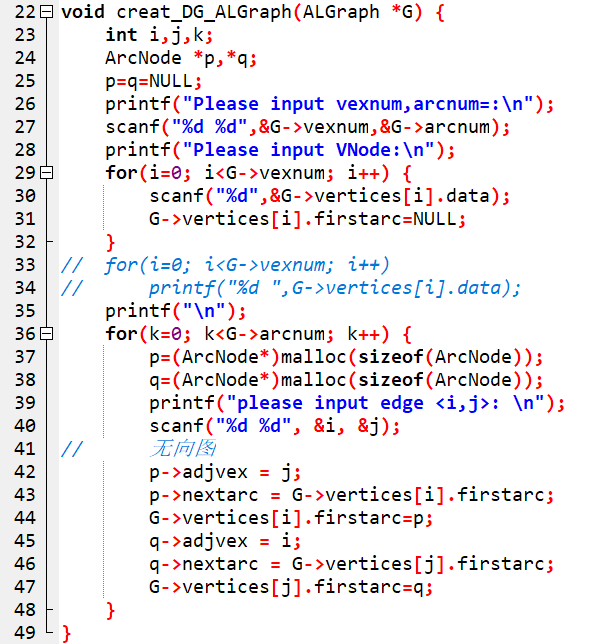
4-2 采用邻接表存储结构，设计一个算法，判别无向图G中指定的两个顶点之间是否存在一条长度为k的简单路径。

注：简单路径是指顶点序列中不含有重复的顶点。

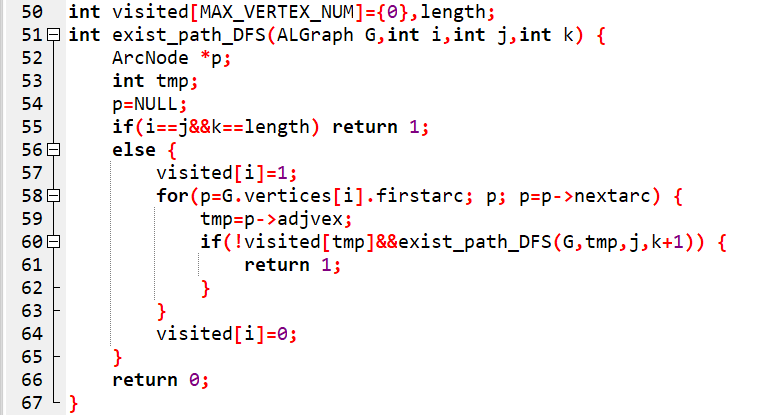
* 编写邻接表存储结构



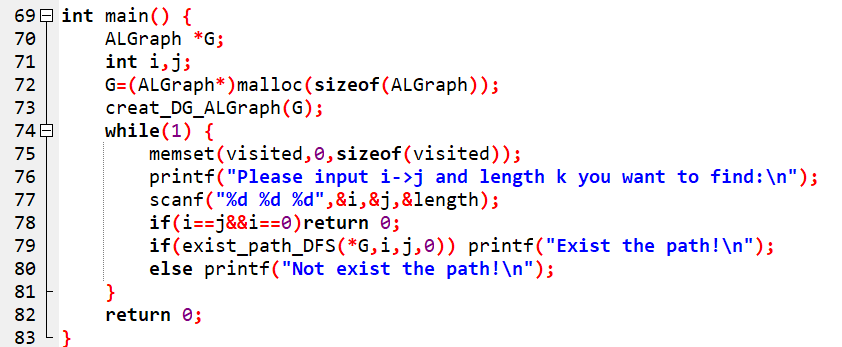
* 创建图 存储为邻接表



* DFS找路径

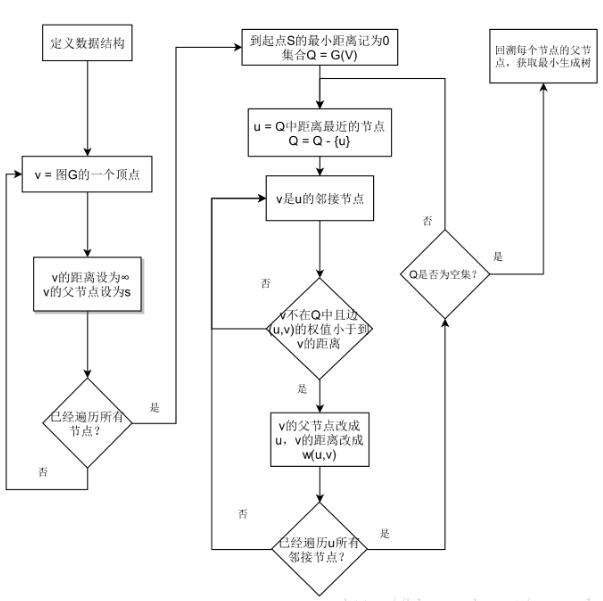


* 主函数交互



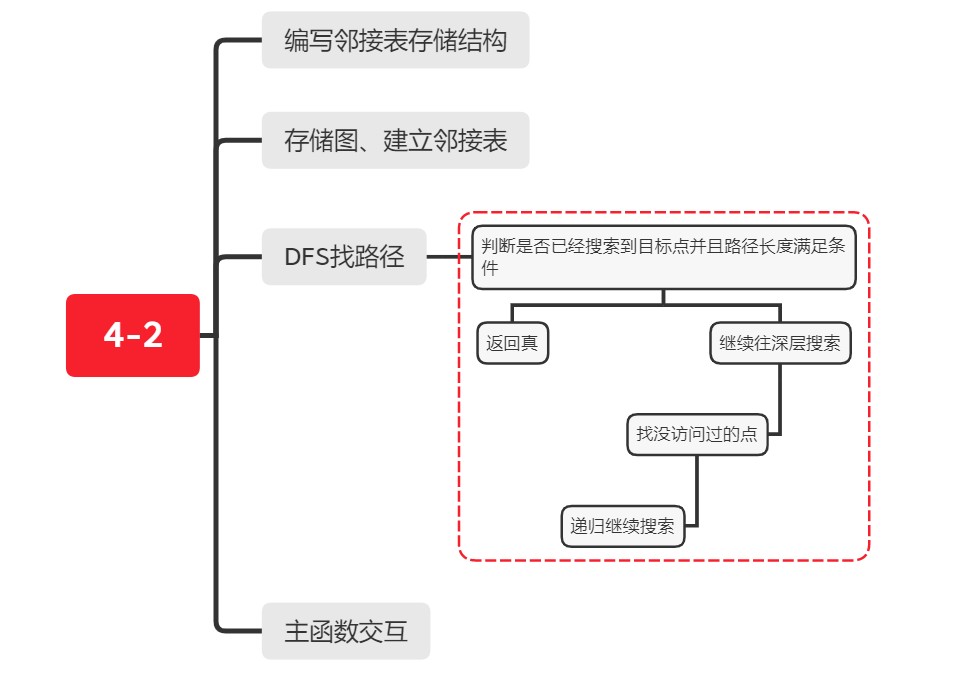
1. **主要算法流程图**

**4-1算法流程**



**（图片太小可见目录下的 4-1算法图.jpg）**

**4-2算法流程**

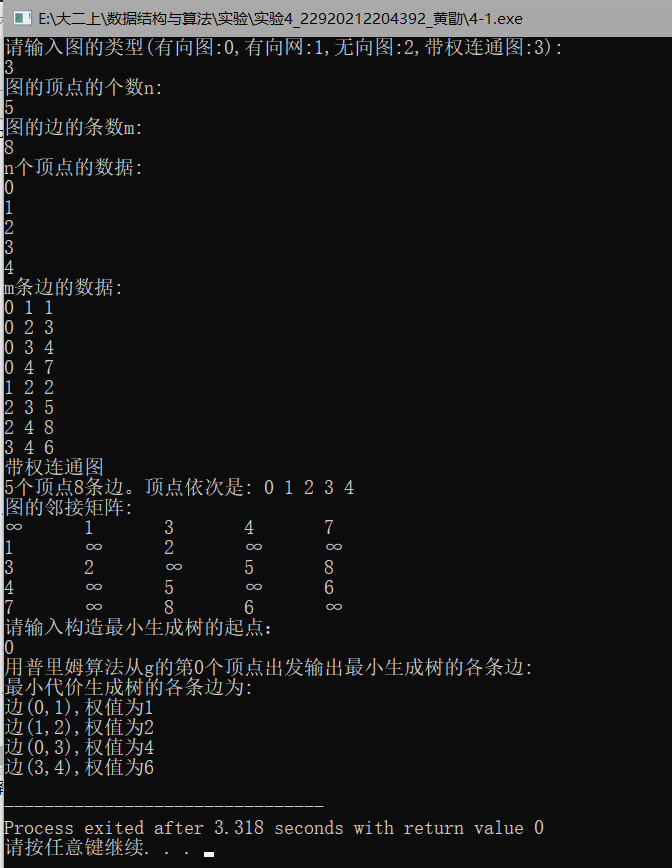


**（可见目录下的 4-2算法图.jpg）**

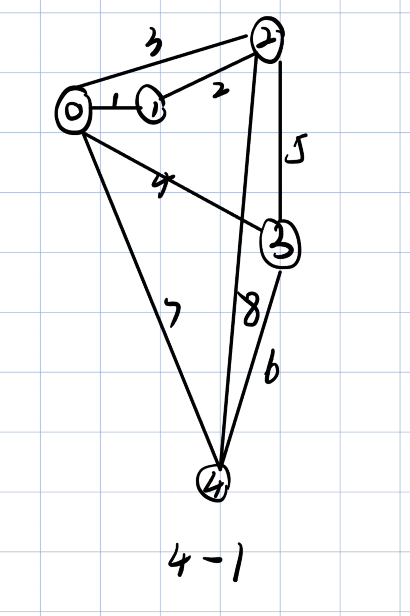
1. **实验结果：**

（结合截图说明算法的输入输出）

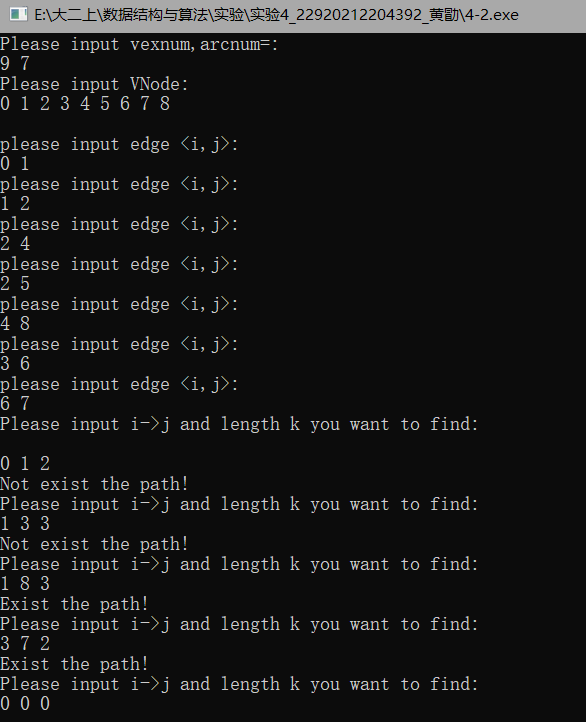
1. 关于4-1的输入与输出：



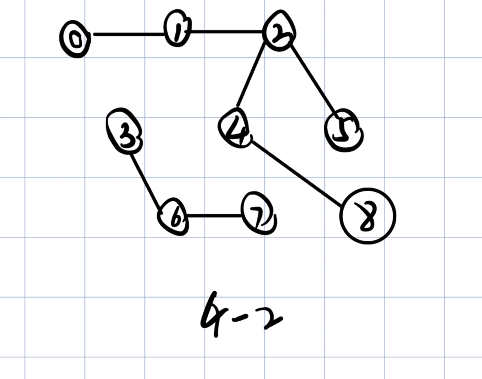
在实际运行中，我创建了如下的图，并生成了正确的最小生成树：



1. 关于4-2的输入与输出



在实际运行中，我创建了如下的图，并找到了正确的路径：



1. **实验小结（即总结本次实验所得到的经验与启发等）：**

在本次实验中，我尝试具体运用了图，在实体机的实验中我能够更深刻地理解对这一部分数据结构的执行方式与特点，并且在编写代码的过程中，我通过不断的调试去寻找语句之间的问题和不足，在潜移默化中提高了我的代码编写能力，这是一次完成效果良好的实验！