**项目说明文档**

**数据结构课程设计**

**——电网建设造价模拟系统**

作 者 姓 名： 戴仁杰

学 号： 1951650

指 导 教 师： 张颖

学院、 专业： 软件学院 软件工程

同济大学

Tongji University

目 录

[1 分析 - 4 -](#_Toc91197345)

[1.1 背景分析 - 4 -](#_Toc91197346)

[1.2 功能分析 - 4 -](#_Toc91197347)

[2 设计 - 4 -](#_Toc91197348)

[2.1 数据结构设计 - 4 -](#_Toc91197349)

[2.2 类结构设计 - 5 -](#_Toc91197350)

[2.3 成员与操作设计 - 5 -](#_Toc91197351)

[2.3.1 Vector类 - 5 -](#_Toc91197352)

[2.3.2 Heap类 - 5 -](#_Toc91197353)

[2.3.3 Edge类 - 6 -](#_Toc91197354)

[2.3.4 Vertex类 - 6 -](#_Toc91197355)

[2.3.5 Graph类 - 7 -](#_Toc91197356)

[2.3.6 PowerGridCostSystem类 - 7 -](#_Toc91197357)

[2.4 系统设计 - 7 -](#_Toc91197358)

[3 实现 - 8 -](#_Toc91197359)

[3.1 创建电网顶点功能的实现 - 8 -](#_Toc91197360)

[3.1.1 创建电网顶点功能流程图 - 8 -](#_Toc91197361)

[3.1.2 创建电网顶点功能核心代码 - 8 -](#_Toc91197362)

[3.1.3 创建电网顶点功能截屏示例 - 9 -](#_Toc91197363)

[3.2 添加电网的边功能的实现 - 9 -](#_Toc91197364)

[3.2.1 添加电网的边功能流程图 - 9 -](#_Toc91197365)

[3.2.2 添加电网的边功能核心代码 - 9 -](#_Toc91197366)

[3.2.3 添加电网的边功能截屏示例 - 10 -](#_Toc91197367)

[3.3 构造最小生成树功能的实现 - 11 -](#_Toc91197368)

[3.3.1 构造最小生成树功能流程图 - 11 -](#_Toc91197369)

[3.3.2 构造最小生成树功能核心代码 - 11 -](#_Toc91197370)

[3.3.3 构造最小生成树功能截屏示例 - 12 -](#_Toc91197371)

[3.4 显示最小生成树功能的实现 - 12 -](#_Toc91197372)

[3.4.1 显示最小生成树功能流程图 - 12 -](#_Toc91197373)

[3.4.2 显示最小生成树功能核心代码 - 13 -](#_Toc91197374)

[3.4.3 显示最小生成树功能截屏示例 - 13 -](#_Toc91197375)

[4 测试 - 14 -](#_Toc91197376)

[4.1 整体功能测试 - 14 -](#_Toc91197377)

[4.2 边界测试 - 14 -](#_Toc91197378)

[4.2.1 电网最小联通时选择构造最小生成树 - 14 -](#_Toc91197379)

[4.2.2 电网全联通时选择构造最小生成树 - 15 -](#_Toc91197380)

[4.3 出错测试 - 15 -](#_Toc91197381)

[4.3.1输入添加顶点或边个数不合法 - 15 -](#_Toc91197382)

[4.3.2 构造最小生成树时起始顶点不在电网中 - 16 -](#_Toc91197383)

[4.3.3 电网未联通时选择构造最小生成树 - 16 -](#_Toc91197384)

1 分析

1.1 背景分析

电在我们的日常生活中占有非常重要的地位，如果没有电，线代城市将无法运转。城市中的电力一般都通过电网来进行运输，而电网的造价成本不菲，在保证全城都能通上电的前提下，如何将电网的造价成本降至最低，是一个非常有经济价值的问题。现在假设一个城市有n个小区，要实现n个小区之间的电网都能够相互接通，构造这个城市n个小区之间的电网，使总工程造价最低。请设计一个能够满足要求的造价方案。

1.2 功能分析

系统的功能要求是在每个小区之间都可以设置一条电网线路，但都要付出相应的经济代价。n个小区之间最多可以有n（n-1）/2条线路，选择其中的n-1条使总的耗费最少。因此首先系统需要小区的布局构造出来，随后在小区之间增加线路。增加完毕后，就需要通过计算来得出其中的n-1条使总的耗费最少，并将它们显示出来。

综上所述，该电网建设造价模拟系统需要有创建电网顶点，添加电网的边，

计算得出n-1条线路使总的耗费最少，输出构造方案的功能。

2 设计

2.1 数据结构设计

一个连通图的每一棵生成树，都是原图的一个极大无环子图。如果每个小区可以看作一个顶点，小区与小区之间的电网线路可以看作一条边，每条线路的造价看作边上的权值，这样就可以把整个电网看作一个连通网络。若一个连通网络由n个顶点组成，则其生成树必含n个结点、n-1条边。由于我们要求建立一个造价最低的电网系统，那就要找出该联通网络的一棵最小生成树。

构造最小生成树的方法最典型的有两种，一种为Kruskal算法，一种为Prim算法。本系统采用了Prim算法构造最小生成树。

Prim算法是在不断迭代进行的。构建两个集合V与V1，V代表当前最小生成树的顶点集合，V1代表不属于当前生成树的顶点集合。首先选定构造最小生成树的起始顶点u0，将它加入到集合V中，随后选择一条边（u，v），要求u属于集合V而v属于集合V1，且该边为满足该条件的权值最小的边。将v从V1取出加入到集合V中，然后继续这个过程，直到网络中的所有顶点都已经加入到生成树顶点集合V中。此时算法过程中选取的边的集合就可以构建出最小生成树。

如上分析所述，该电网造价模拟系统会有大量创建顶点，添加边的操作。但边不会很密集，因此采用用邻接表表示的图的数据结构来保存所有小区与线路。由于Prim算法每次都要获得权值最小的边，因此设计了一个最小堆的数据结构，每次从堆顶取出权值最小的边。

2.2 类结构设计

首先，系统运行过程中的数据主要是通过Vector向量类来保存。其次，本系统有一个图（Graph）类，相对应的有图中所需的顶点（Vertex）类和边（Edge）类。同时，为了给Prim算法提供最小权值的边，本系统设计了一个堆（Heap）类。最后，本系统还设计了电网造价系统（PowerGridCostSystem）类，给用户提供添加顶点，添加边，构建或显示最小生成树等功能的接口。为了使数据结构更具有泛用性，本系统将Vector类，Heap类，Vertex类，Edge类，Graph类等都设计为了模板类。

2.3 成员与操作设计

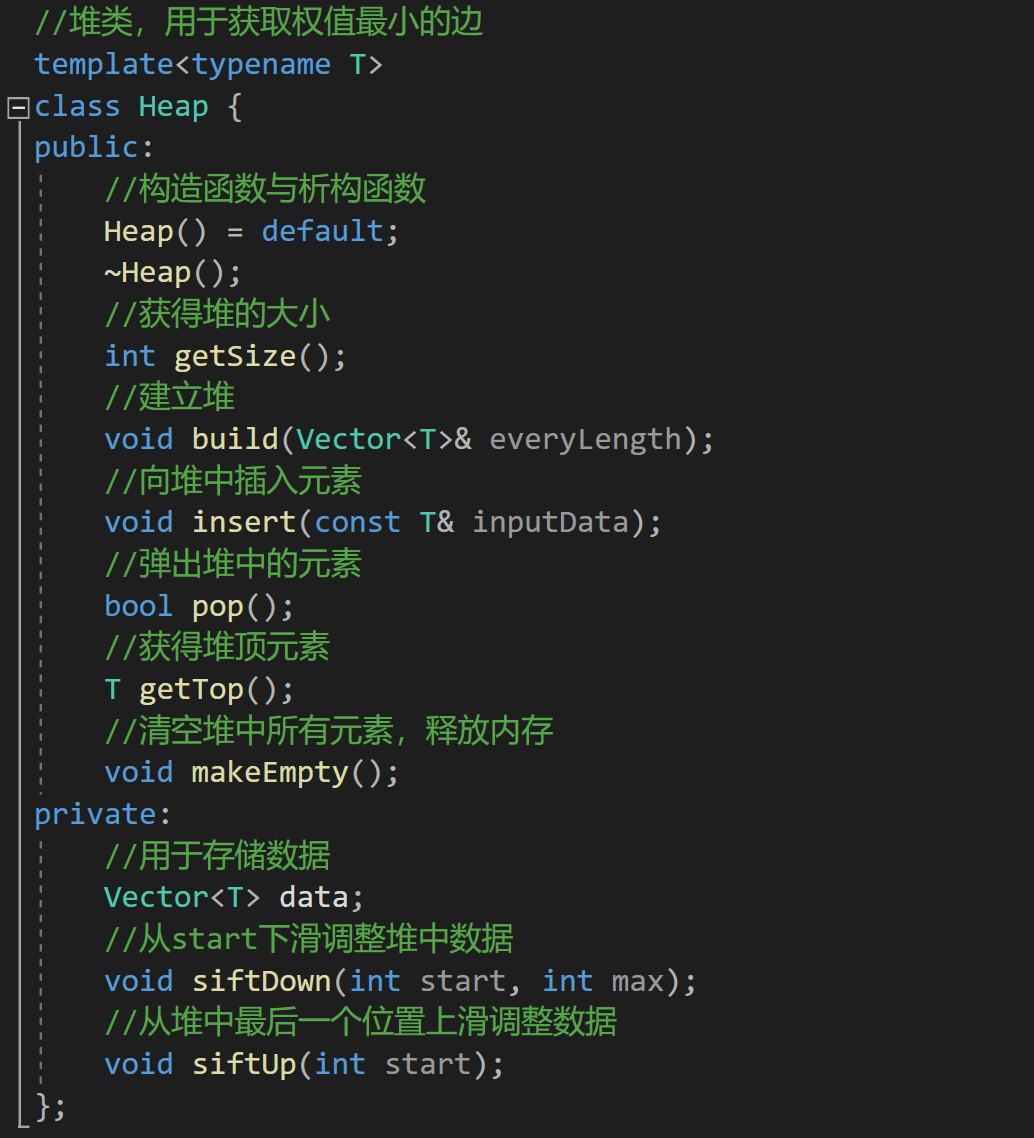
2.3.1 Vector类

Vector类为向量类，采用了类模板提高泛用性。类内实现了下标索引，插入元素，删除元素，获得数组大小等操作。通过动态内存申请调整大小，避免直接声明数组造成的空间浪费。



2.3.2 Heap类

Heap类为堆类，采用了类模板提高泛用性。类内实现了堆的增删元素，堆的上滑下滑调整、获取堆顶元素和清空堆操作。类内成员为一个Vector类型的向量数组元素。



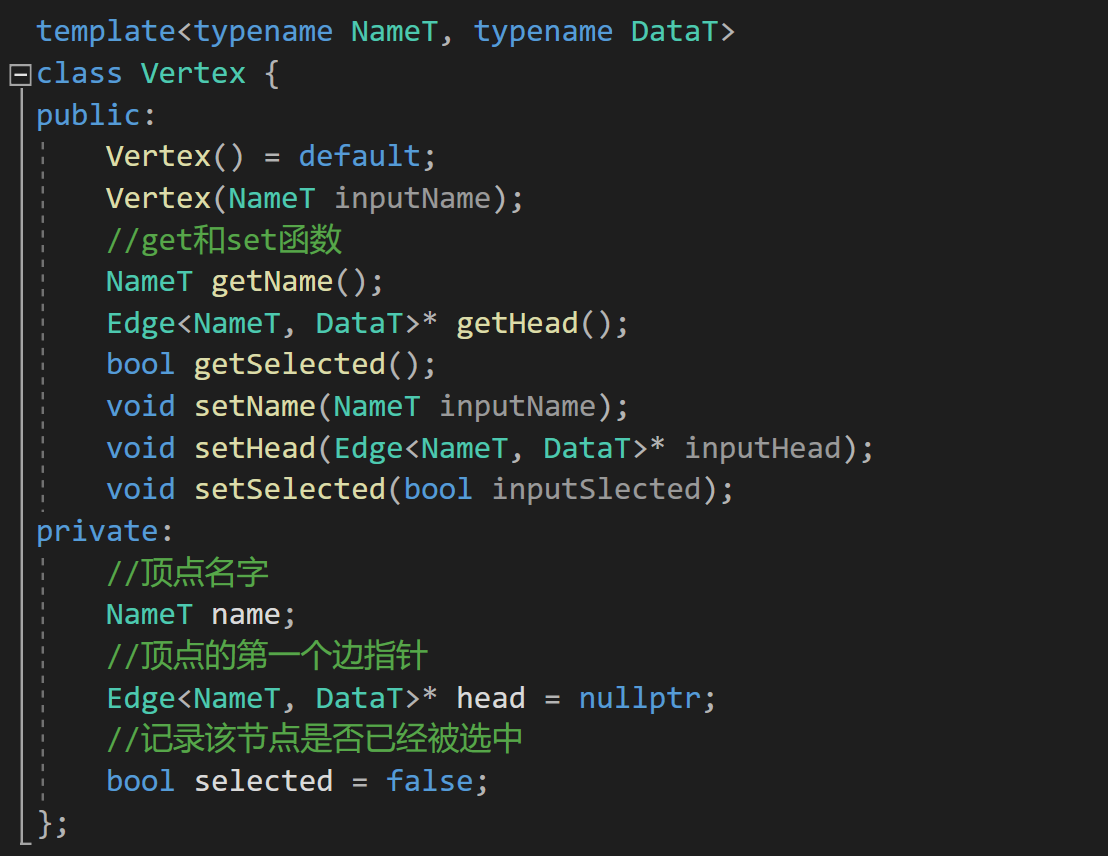
2.3.3 Edge类

Edge类为边类，采用了类模板提高泛用性。提供默认构造函数和赋值构造函数创建边。并通过一系列get与set函数与外界交流。类内成员为边的两个顶点、边的权重以及指向下一条边的指针。



2.3.4 Vertex类

Vertex类为顶点类，采用了类模板提高泛用性。提供默认构造函数和赋值构造函数创建顶点。并通过一系列get与set函数与外界交流。类内成员为顶点名字、顶点的第一条边指针以及顶点是否被选中的标记。



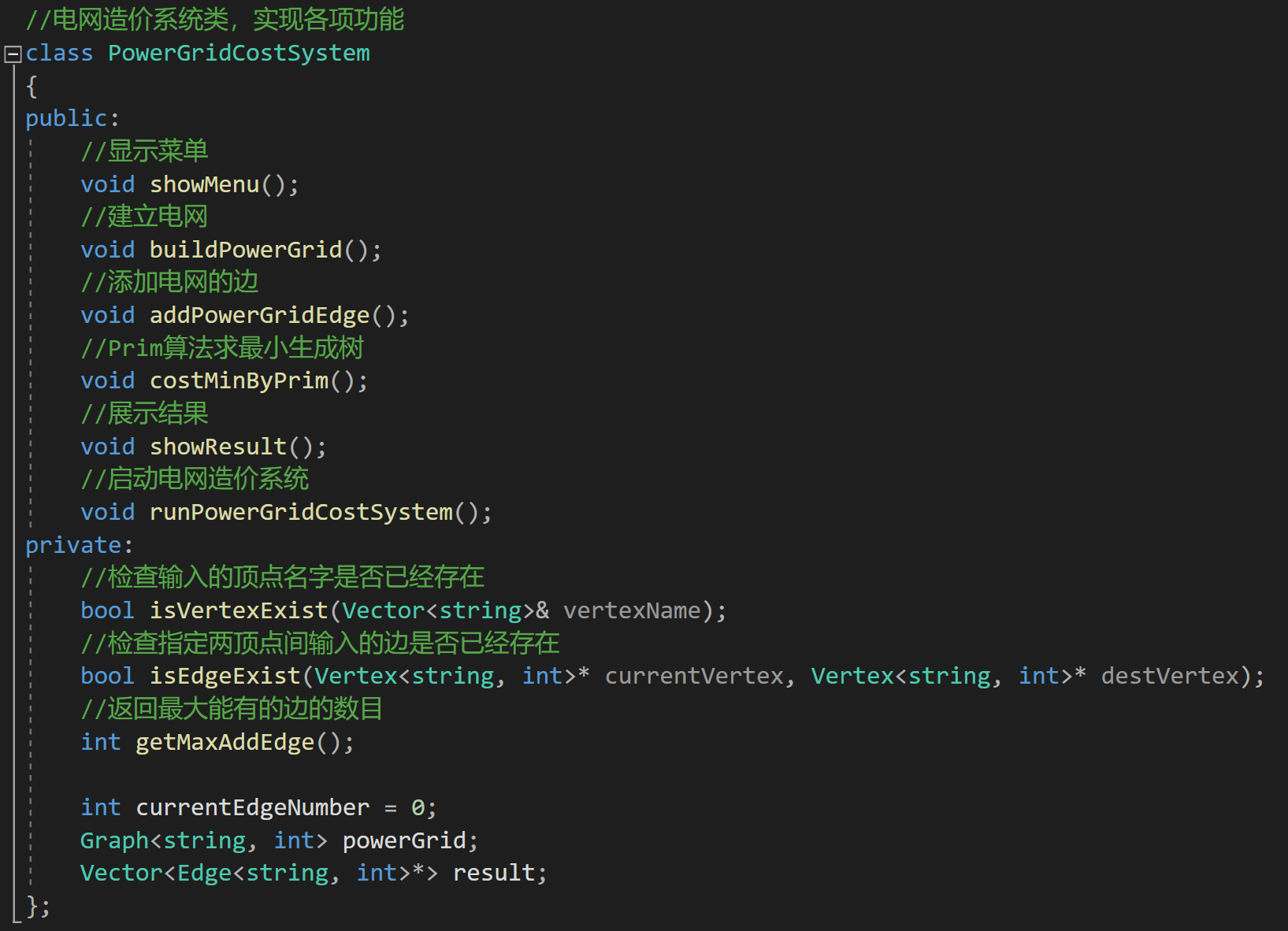
2.3.5 Graph类

Graph类为图类，采用了类模板提高泛用性。类内实现了图的插入顶点、插入边、寻找顶点、获取顶点表和清空堆操作。类内成员为一个Vector类型的向量数组元素，表示顶点表的信息。



2.3.6 PowerGridCostSystem类

PowerGridCostSystem类为电网造价系统类。对图类进行封装增强，丰富了交互提示信息，增强了输入错误、边界错误处理性能。类内封装了显示菜单、建立电网、添加电网的边、Prim算法求最小生成树、展示结果等函数。并封装了运行家谱管理系统的函数runPowerGridCostSystem ()，增强了外部调用的友好性。类内数据为一个存储电网信息的图、存储当前边数的整数和一个存放最小生成树结果的Vector数组。



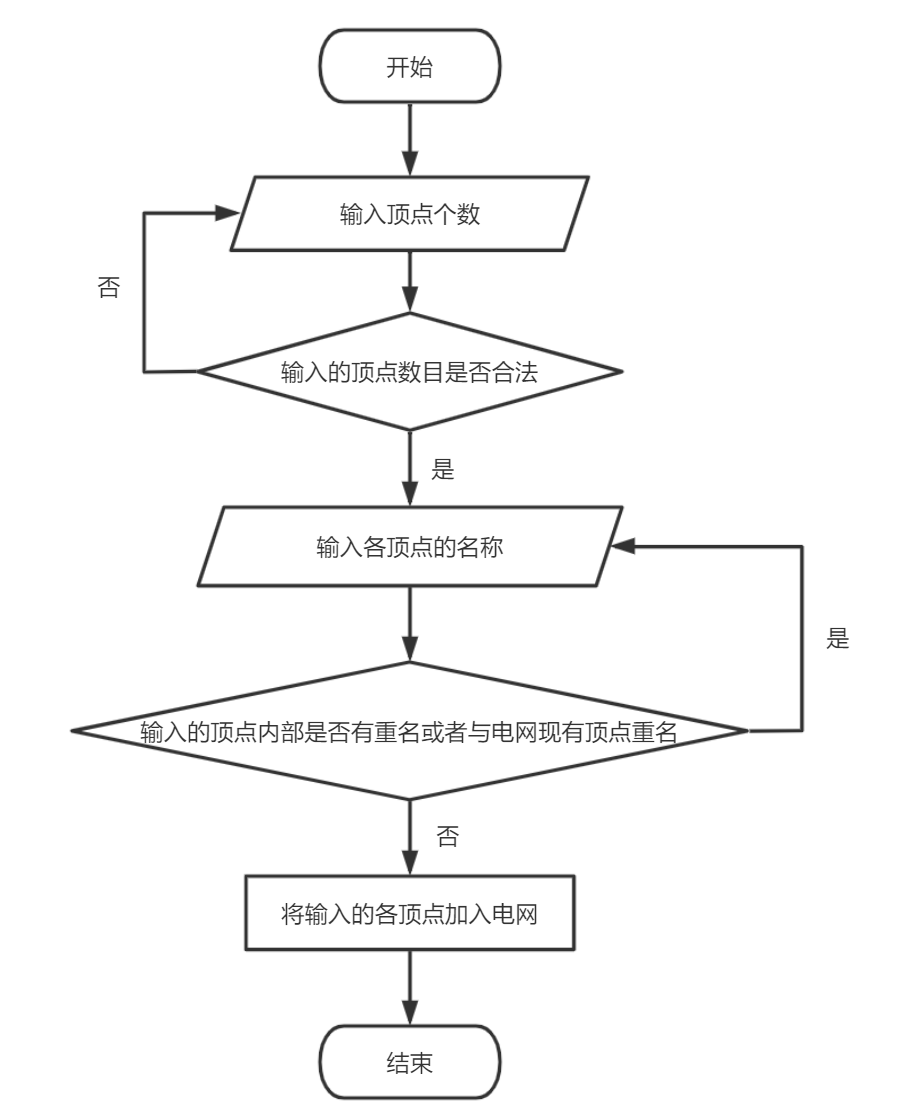
2.4 系统设计

系统首先会在屏幕上显示一个用户操作菜单，根据此菜单，用户通过输入对应的操作码来执行相应的操作。功能有创建电网顶点，添加电网的边，构造最小生成树，显示最小生成树，退出程序等等。

3 实现

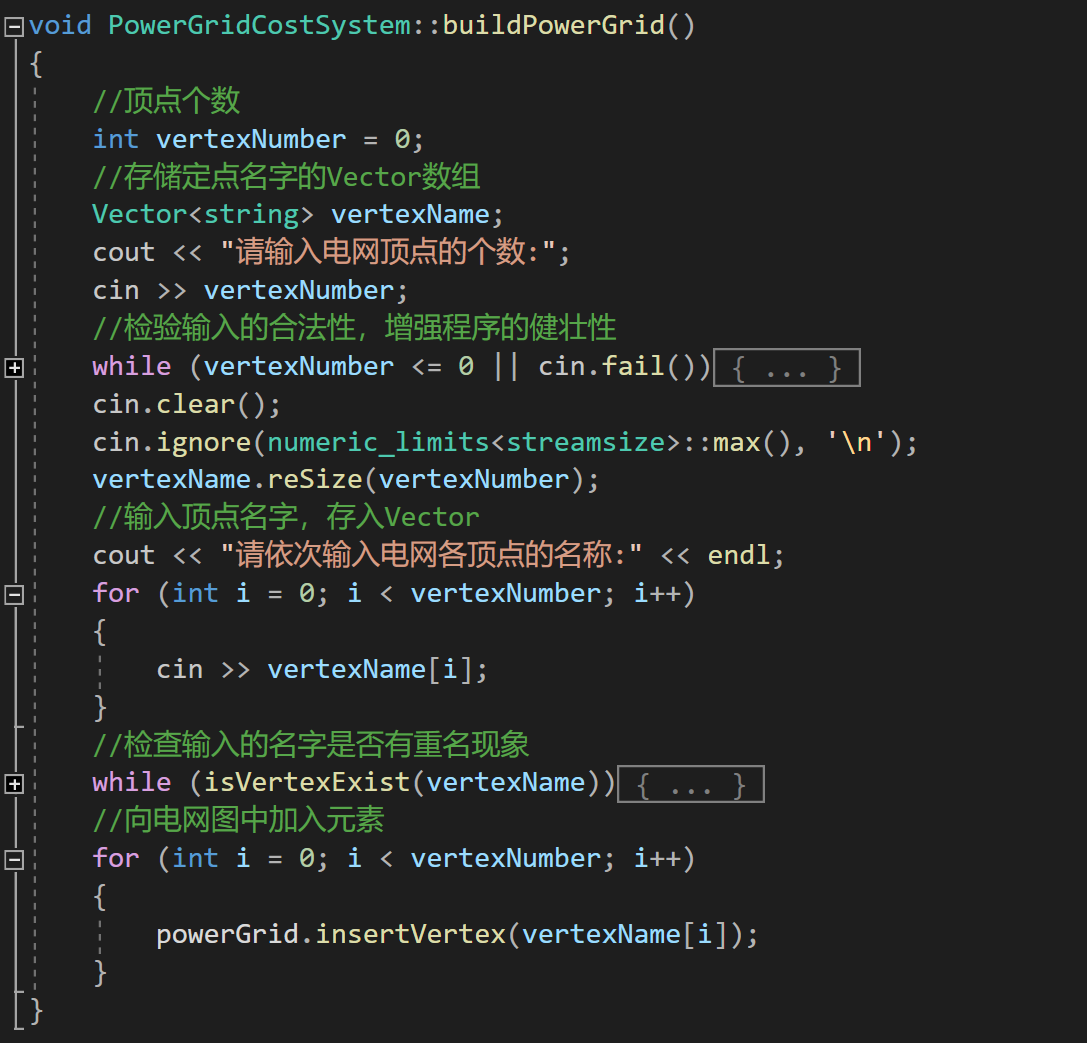
3.1 创建电网顶点功能的实现

3.1.1 创建电网顶点功能流程图

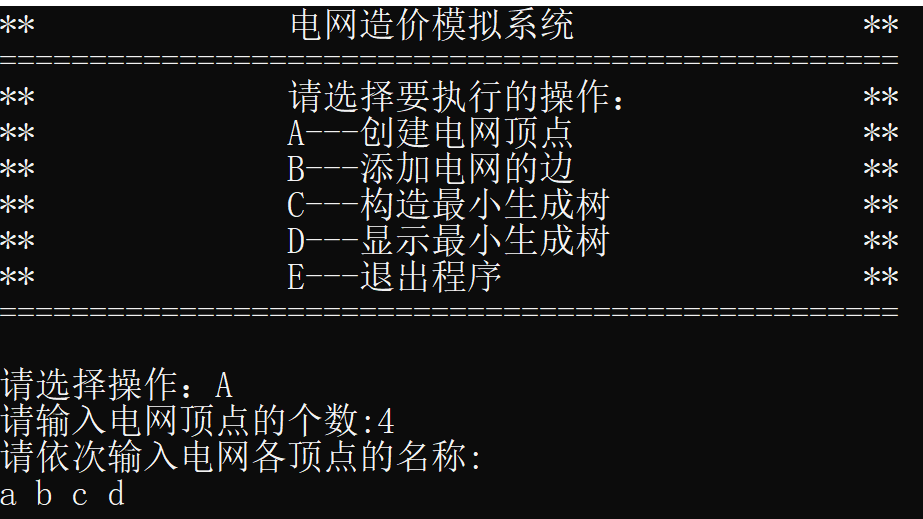


3.1.2 创建电网顶点功能核心代码

PowerGridCostSystem类中的buildPowerGrid()函数(折叠了部分错误处理代码)

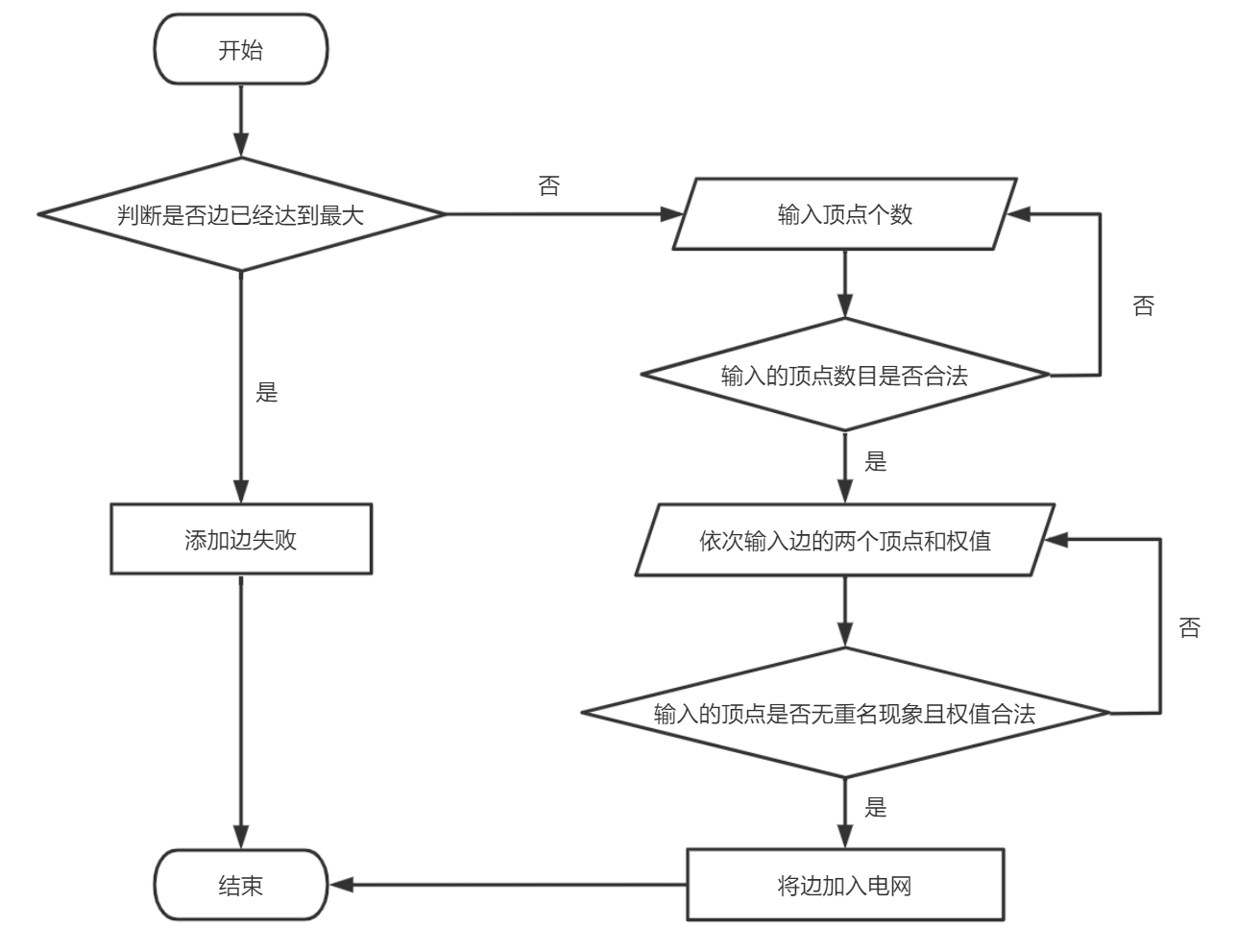


3.1.3 创建电网顶点功能截屏示例



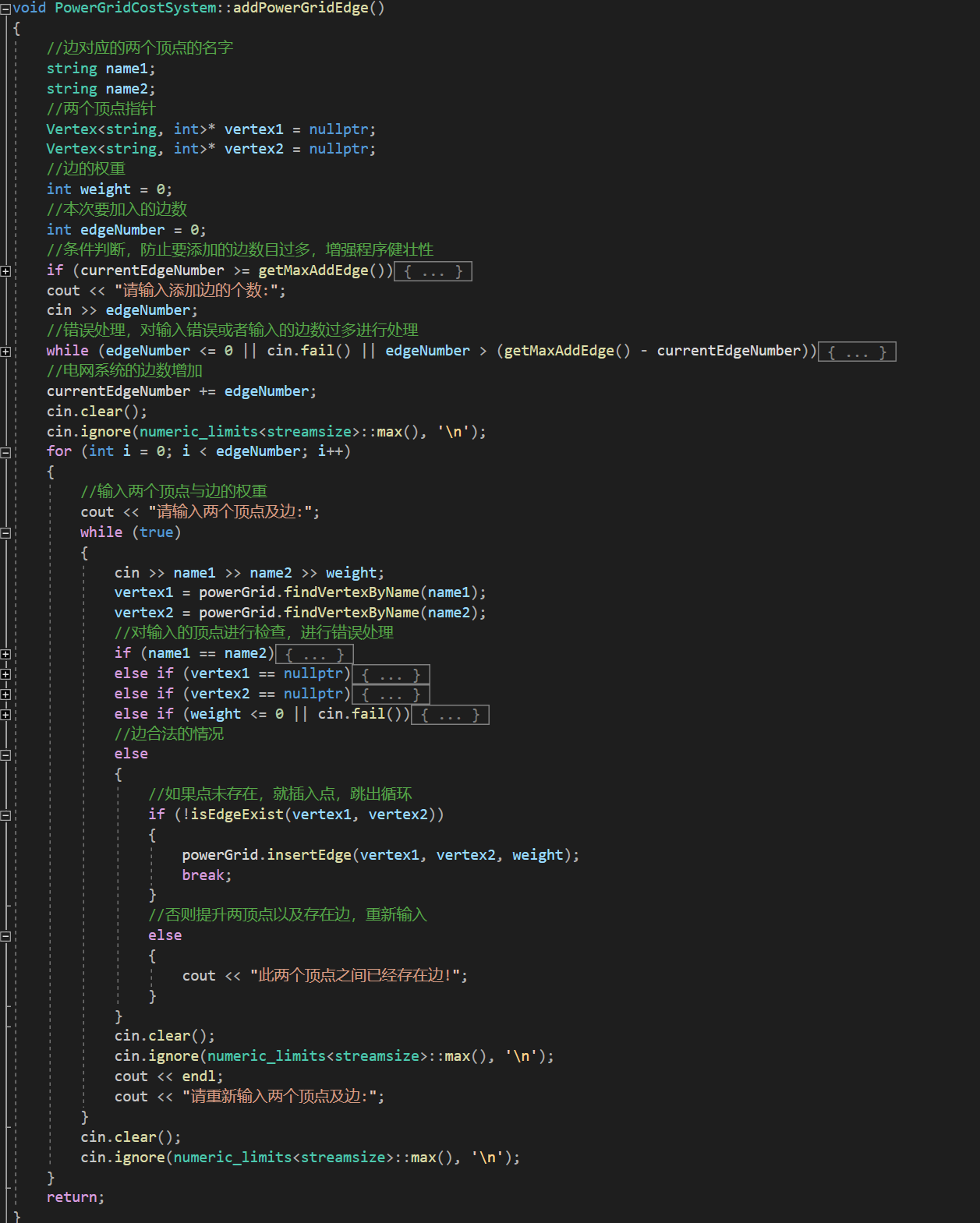
3.2 添加电网的边功能的实现

3.2.1 添加电网的边功能流程图

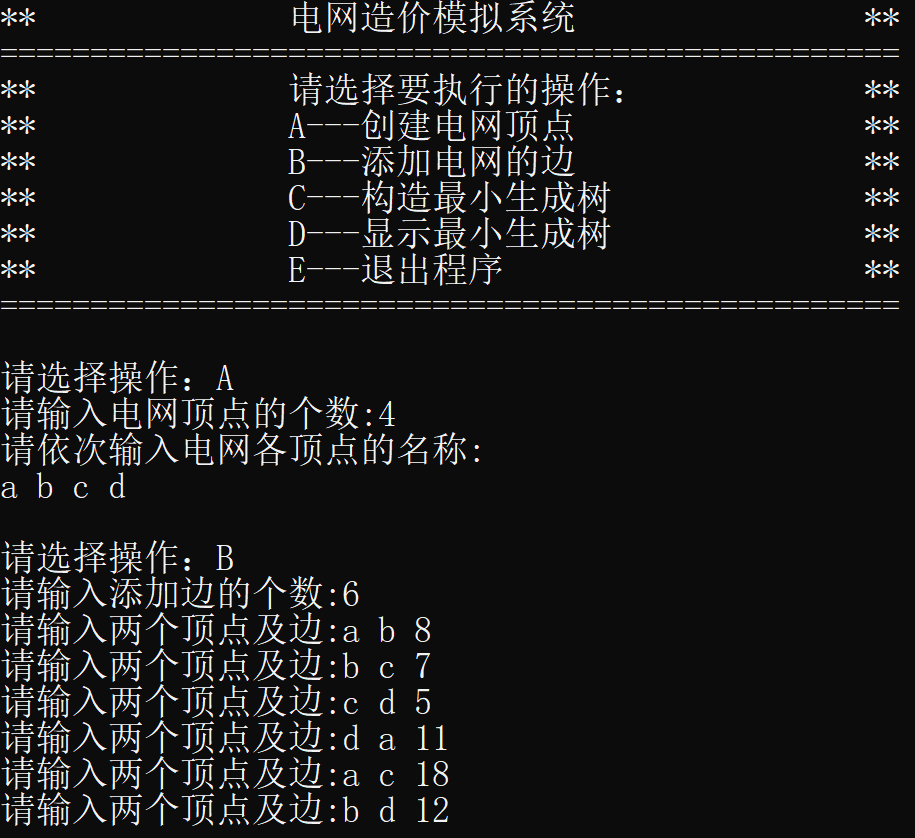


3.2.2 添加电网的边功能核心代码

PowerGridCostSystem类中的addPowerGridEdge()函数(折叠了部分错误处理代码)

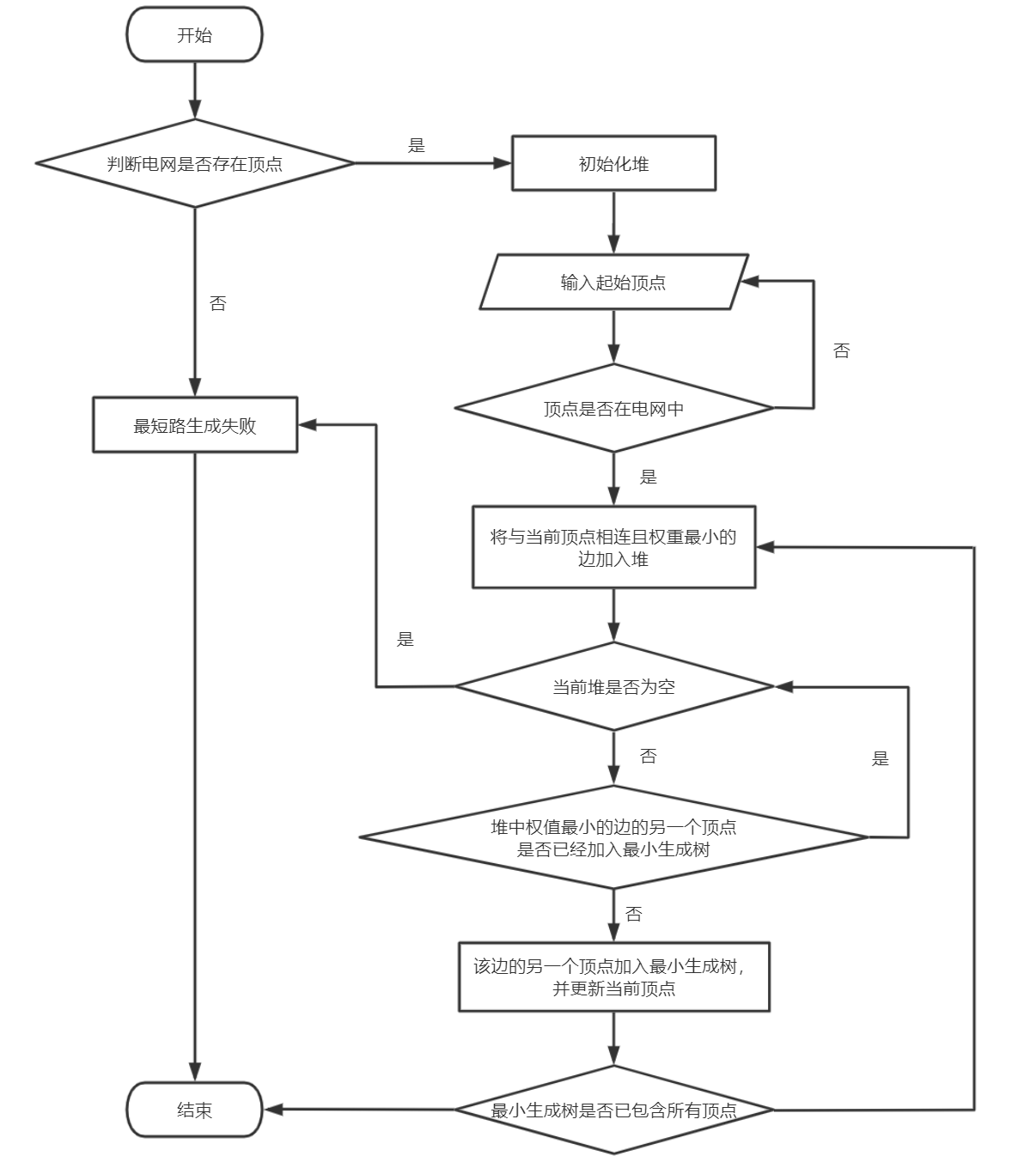


3.2.3 添加电网的边功能截屏示例



3.3 构造最小生成树功能的实现

3.3.1 构造最小生成树功能流程图

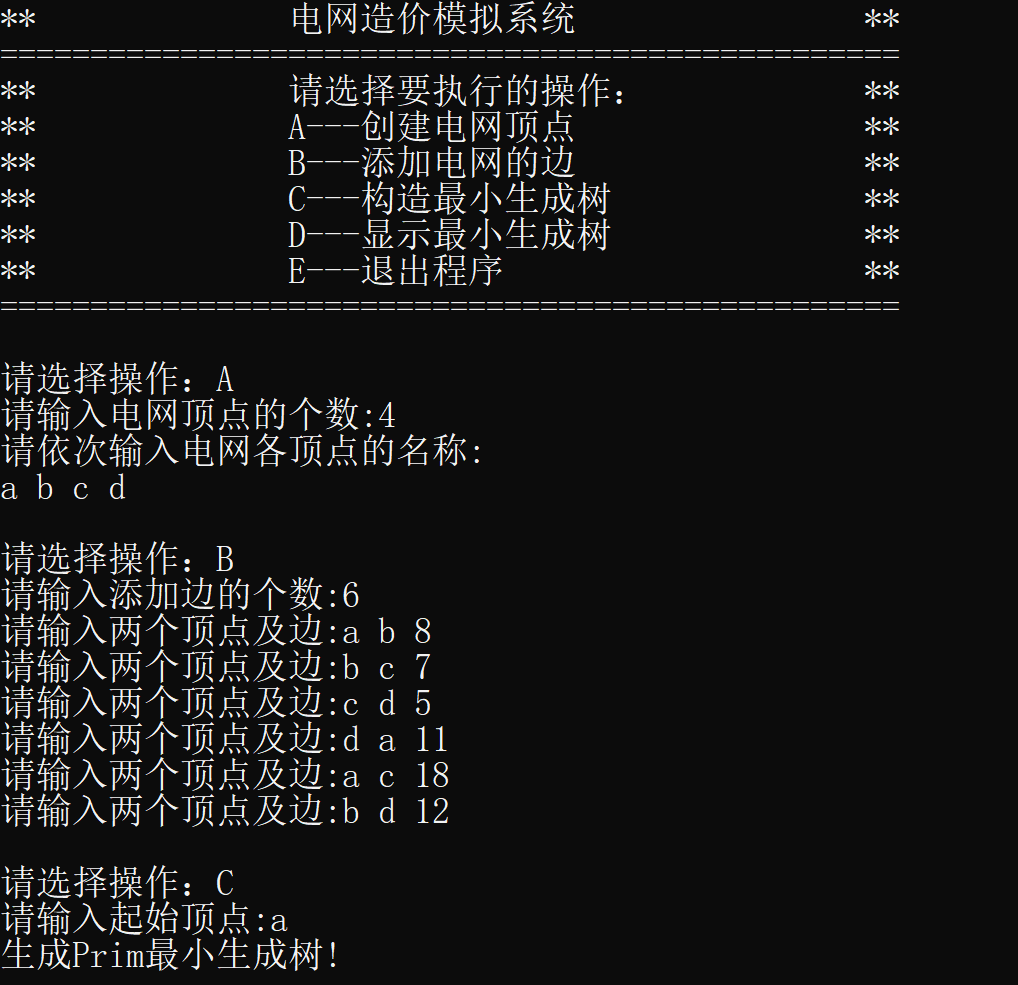


3.3.2 构造最小生成树功能核心代码

PowerGridCostSystem类中的costMinByPrim ()函数(仅截取部分)

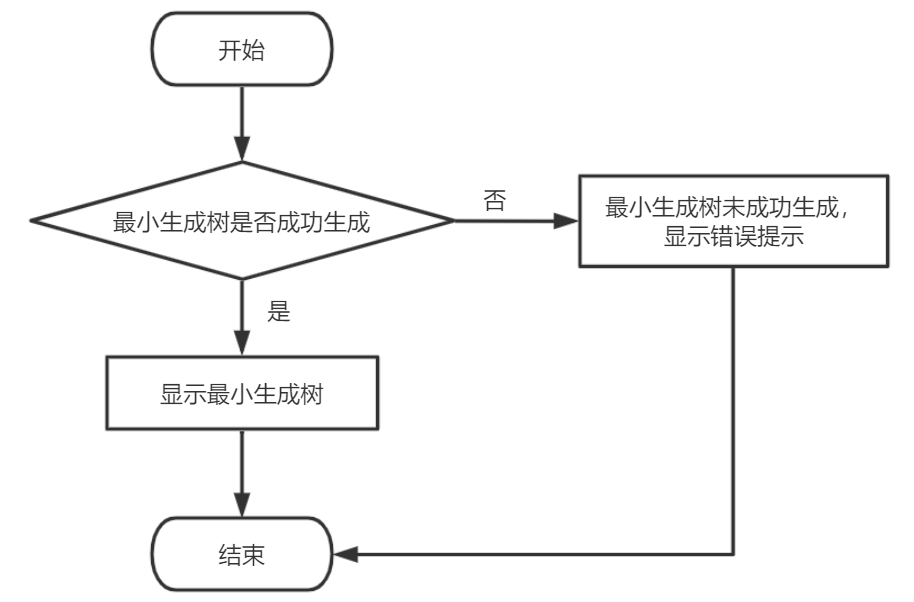


3.3.3 构造最小生成树功能截屏示例



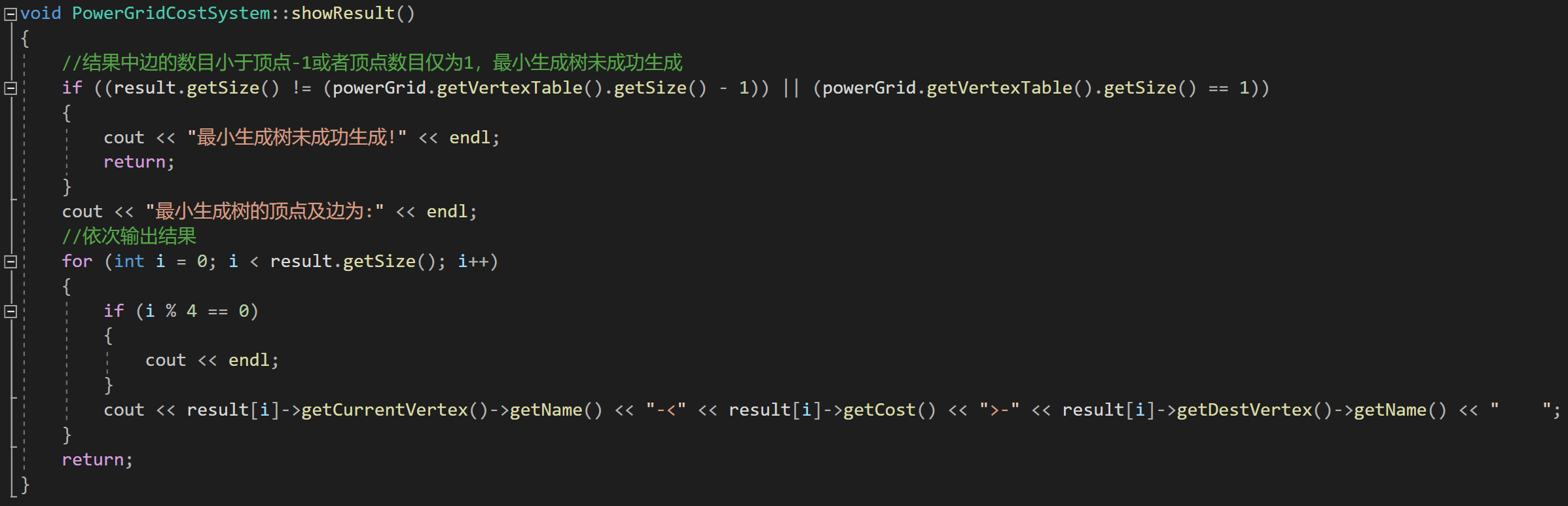
3.4 显示最小生成树功能的实现

3.4.1 显示最小生成树功能流程图

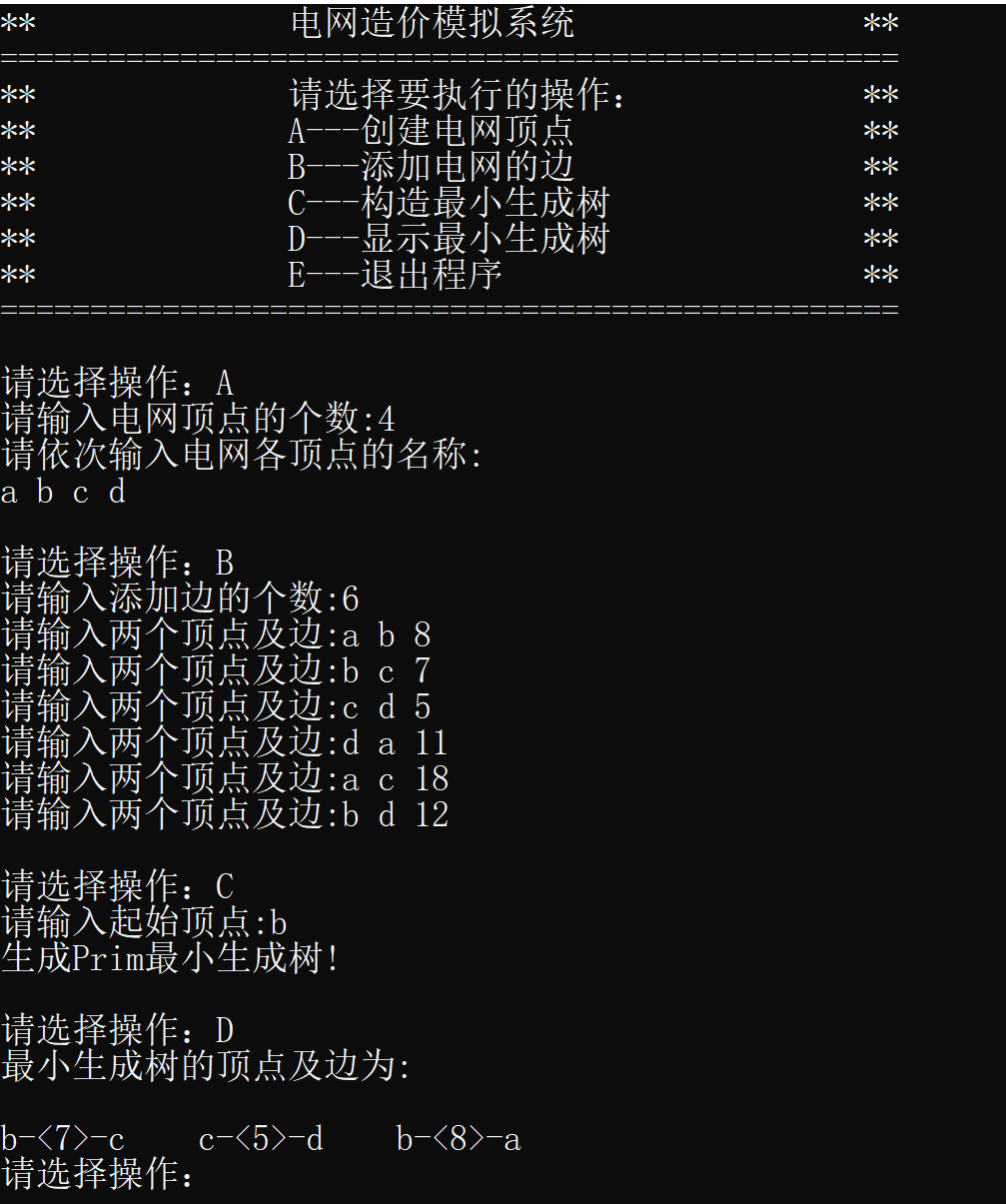


3.4.2 显示最小生成树功能核心代码

PowerGridCostSystem类中的showResult ()函数



3.4.3 显示最小生成树功能截屏示例

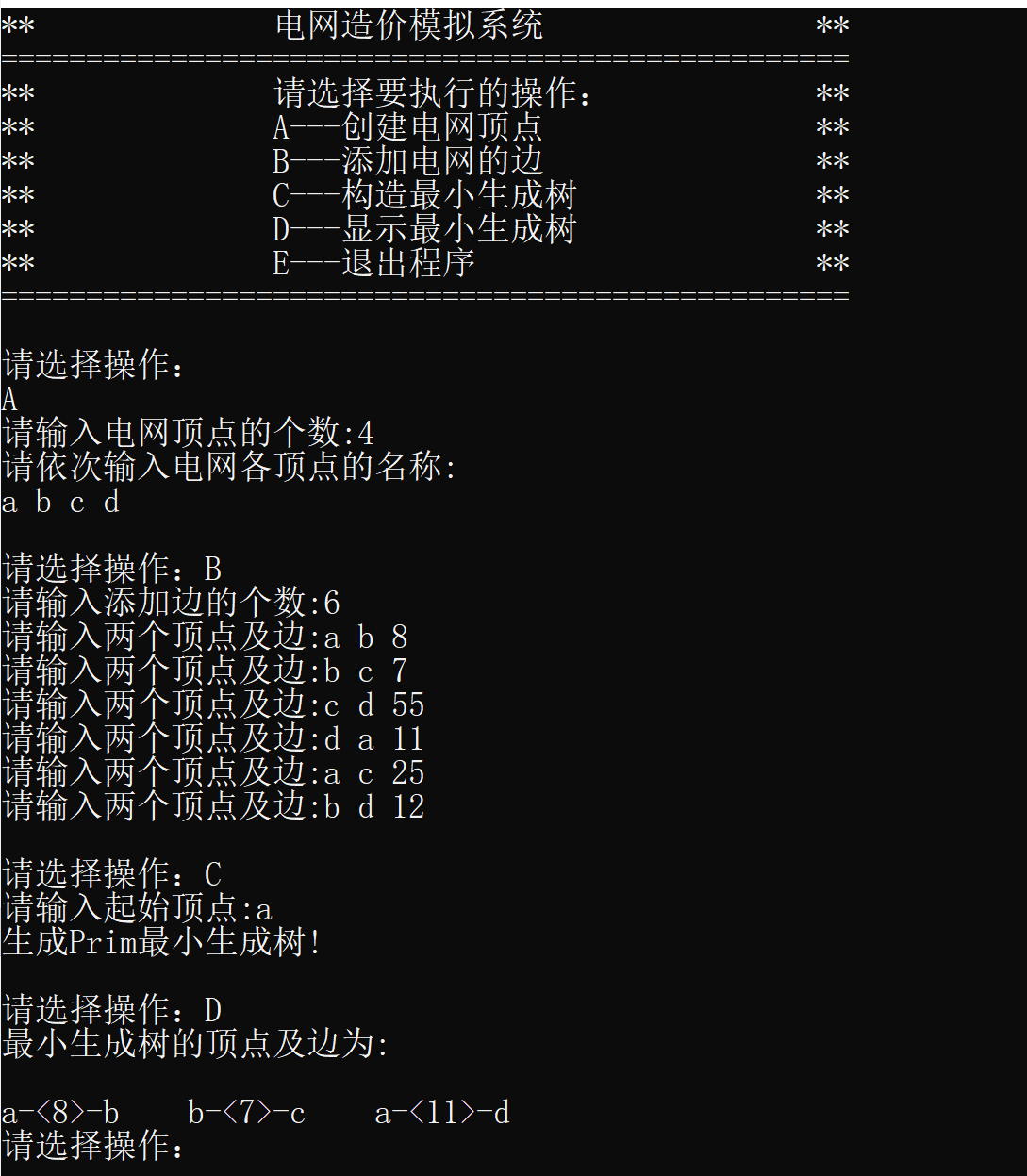


4 测试

4.1 整体功能测试

**测试用例：**4个顶点6条边产生最小生成树

**实验结果：**结果正确

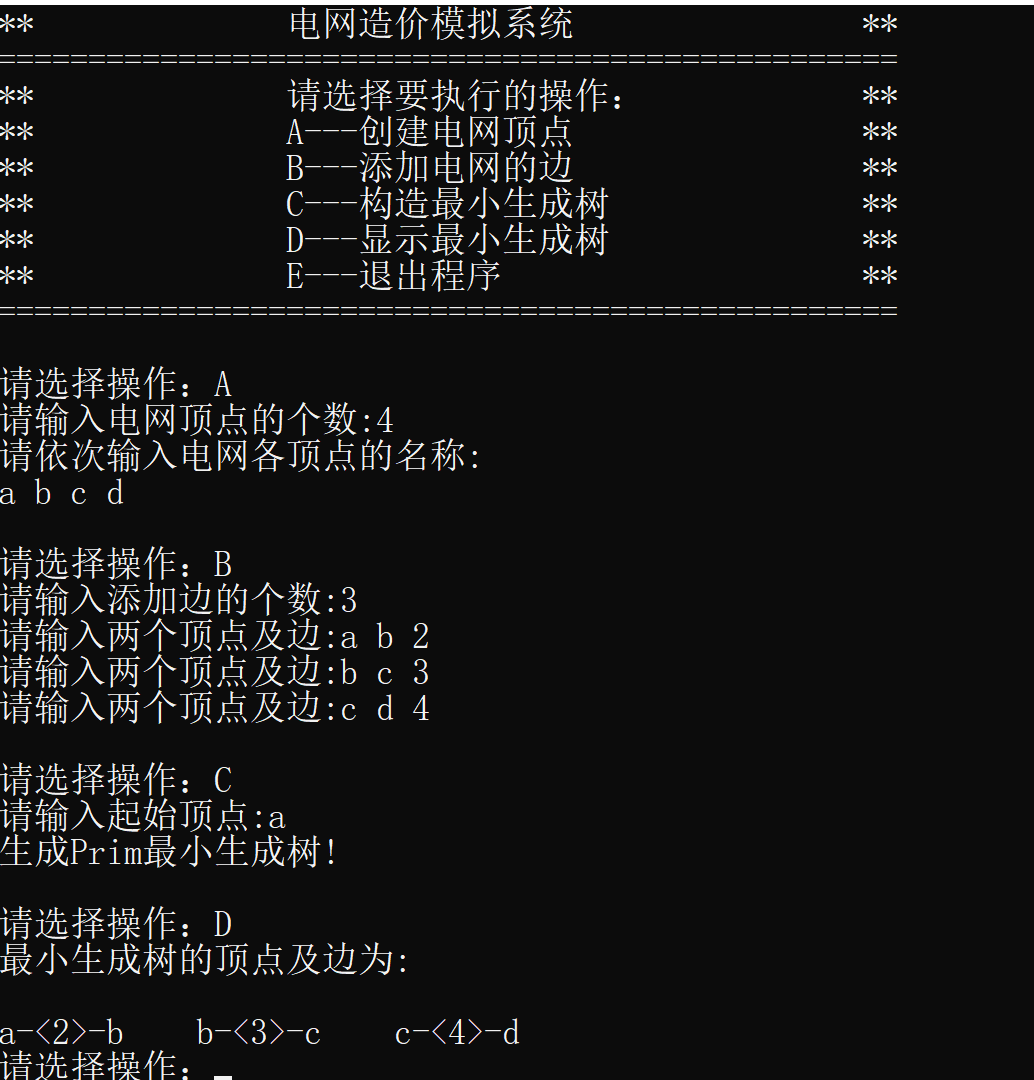


4.2 边界测试

### 4.2.1 电网最小联通时选择构造最小生成树

**测试用例：**4个顶点3条边联通

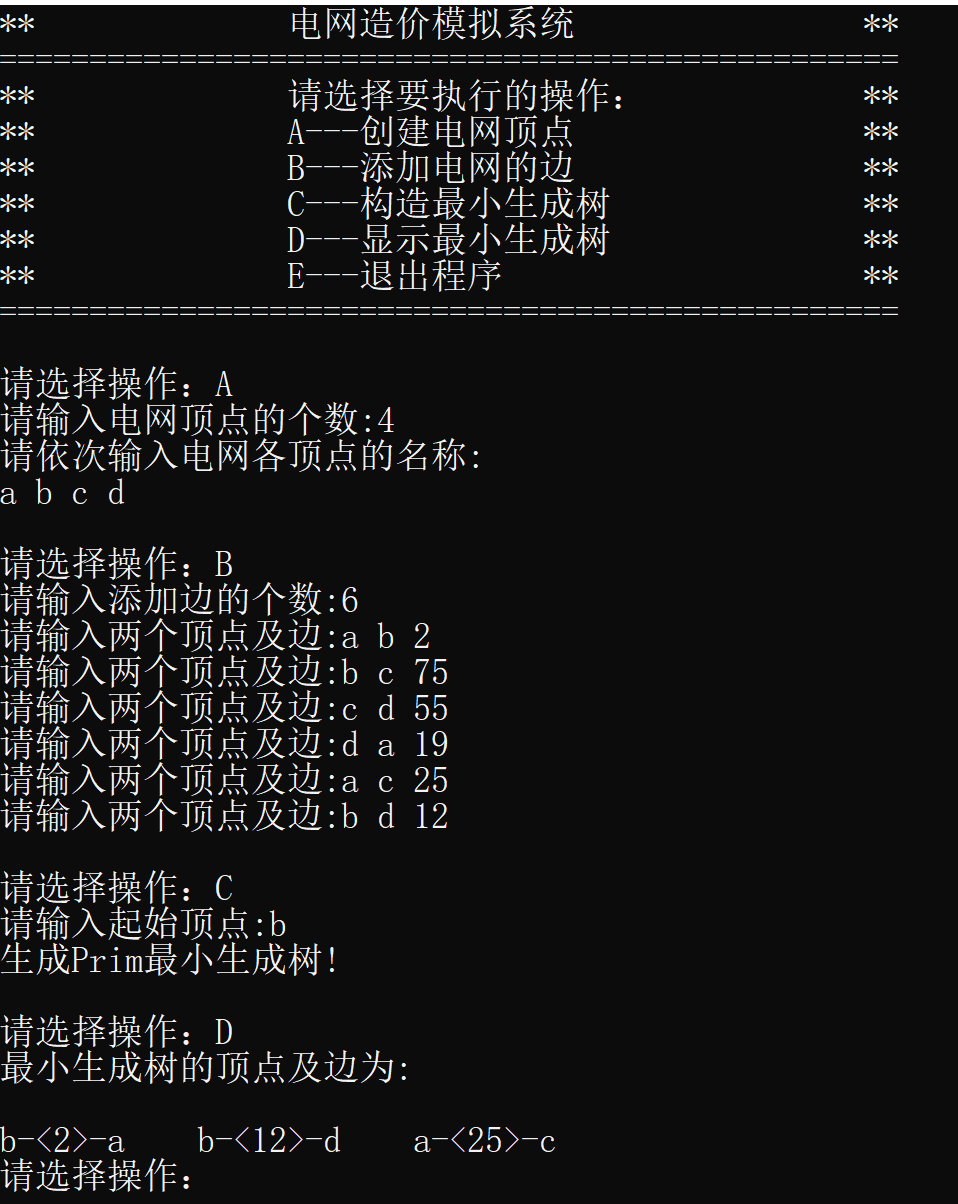
**实验结果：**结果正确



### 4.2.2 电网全联通时选择构造最小生成树

**测试用例：**4个顶点6条边联通

**实验结果：**结果正确



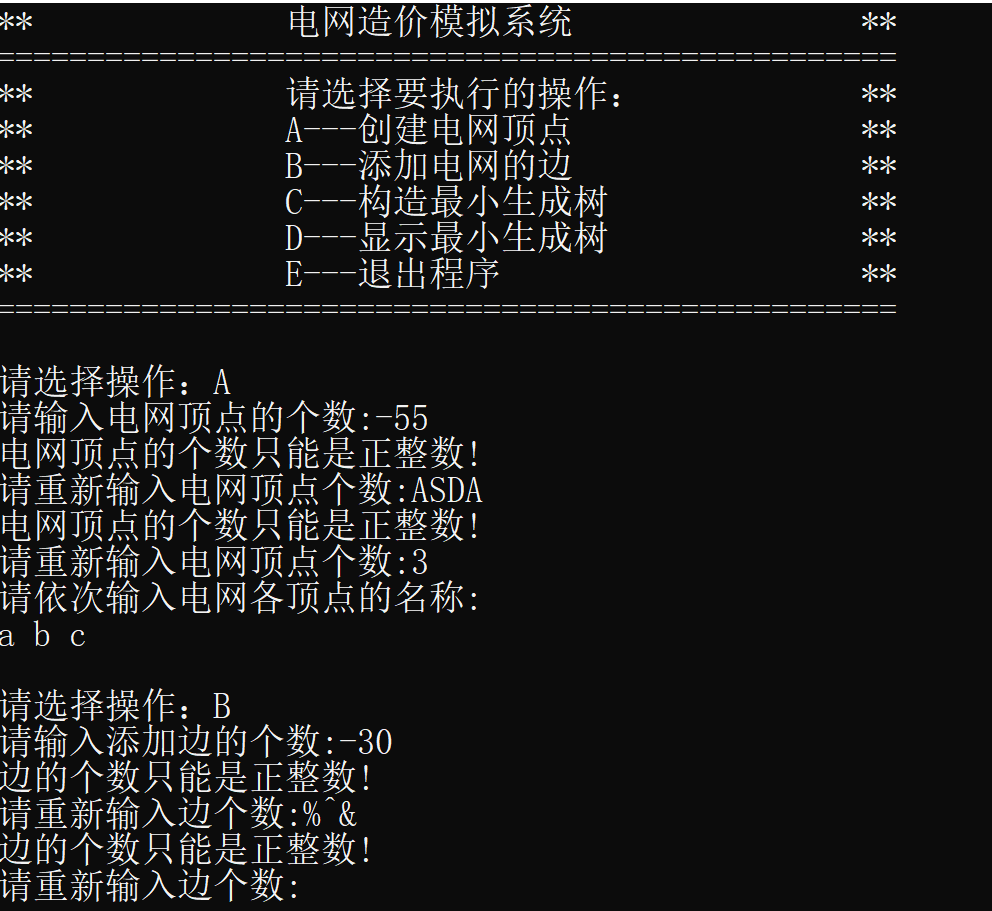
4.3 出错测试

4.3.1输入添加顶点或边个数不合法

**测试用例：**输入的数字为负数或非数字

**预期结果：**程序给出提示信息，程序正常运行不崩溃。

**实验结果：**

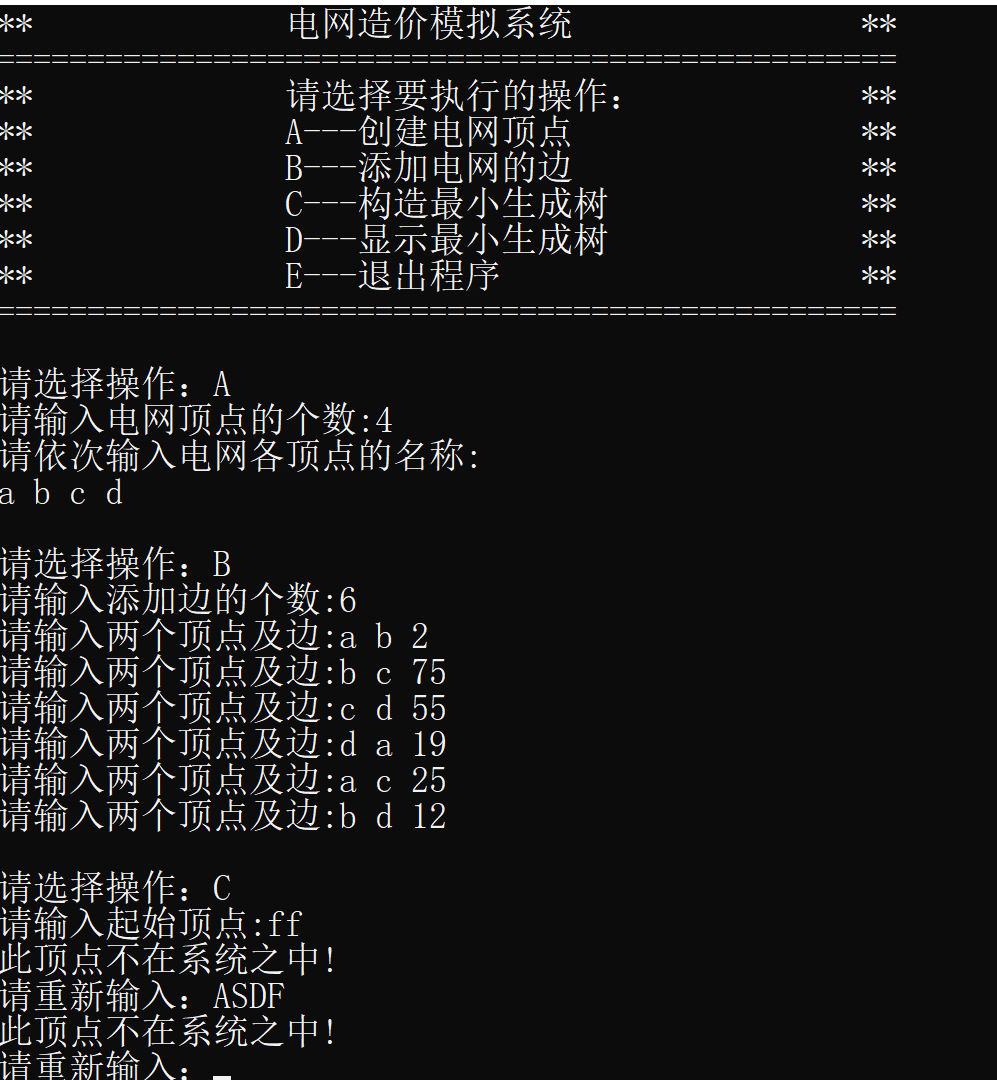


4.3.2 构造最小生成树时起始顶点不在电网中

**测试用例：**顶点为不在电网中的点

**预期结果：**程序给出提示信息，程序正常运行不崩溃。

**实验结果：**



4.3.3 电网未联通时选择构造最小生成树

**测试用例：**4个顶点2个边，未联通

**预期结果：**程序给出提示信息，程序正常运行不崩溃。

**实验结果：**

