

目录

第一部分 算法实现设计说明	3
1.1 题目	3
1.2 软件功能	3
1.3 设计思想	4
1.4 逻辑结构与物理结构	6
1.5 开发平台	6
1.6 系统的运行结果分析说明	7
1.7 操作说明	10
第二部分 综合应用设计说明	15
2.1 题目	15
2.2 软件功能	15
2.3 设计思想	17
2.4 逻辑结构与物理结构	18
2.5 开发平台	18
2.6 系统的运行结果分析说明	18
2.7 操作说明	24
第三部分 实践总结	29
3.1 所做的工作	29
3.2 总结与收获	29
第四部分 参考文献	31

第一部分 算法实现设计说明

1.1 题目

以邻接矩阵确定一个图，完成：

- (1) 建立并显示出它的邻接链表；
- (2) 以递归及非递归的方式进行深度优先遍历，显示遍历的结果，并且随时显示栈的入、出情况。
- (3) 对该图进行广度优先遍历，显示遍历的结果，并随时显示队列的入、出情况。

1.2 软件功能

(1) 用户界面菜单栏

File菜单下设置EXIT选项，点击之后退出程序界面。

实现方式：

Qt槽函数和信号连接机制，即点击EXIT选项（产生信号）之后进行触发槽函数，实现相应的动作。

(2) 创建邻接矩阵

此处只需用户选择相应的矩阵大小（矩阵大小有一个范围限制，例如10*10为最大），再选择相应的要建立的图种类，点击生成邻接矩阵按钮即可生成一个两个维度相同的邻接矩阵。

实现方式：

通过矩阵选择菜单MatrixSize和图种类选择菜单GraphKind, 直接在下拉菜单项中选择相应内容即可。通过QComboBox类实现；

创建邻接矩阵按钮时触发设置的槽函数，将QComboBox中的矩阵维度和图种类传回MGraph类中的CreatGraph()函数进行创建。为了减少用户直接输入邻接矩阵的麻烦，直接生成一个随机数组（01序列），其大小是——维度平方/2-维度大小，这样就能根据图种类进行矩阵赋值生成处理。

(3) 生成邻接链表

用户通过点击设置的邻接链表生成按钮即可在下面的文本框中进行实时展示；

实现方式：

通过前面一步创建的邻接矩阵，即可在点击按钮时触发槽函数，调用MGraph类中的DisplayALGraph()函数将输出的邻接链表用QString进行连接，再用QTextBrowser显示即可。

(4) 生成图

用户通过创建的邻接矩阵，维度表示图的顶点数，一二维组成的数对结果（0表示连接，1表示不相连）形成图形化界面的“图”，在相应的区域显示即可；

实现方式：

通过QPainter类的drawEllipse()和drawText()函数画圆形节点和编号标记，及drawLine()函数画线进行两个节点是否相连的标识。

(5) 广度优先遍历（BFS）

用户通过点击广度优先遍历QGroupBox中的“非递归”按钮，即可在下方的QTextBrowser文本框中进行显示入、出队列情况及遍历的结果。

实现方式：

Qt的槽函数和信号机制，进行点击事件的响应，再调用MGraph类中的BFSTraverse函数进行广度优先遍历即可，将每次入队、出队情况用QString记录，再显示即可。

(6) 深度优先遍历（DFS）

用户通过点击深度优先遍历QGroupBox中的“非递归”和“递归”按钮，即可在下方相应QTextBrowser文本框中显示入栈、出栈情况及结果。

实现方式：

Qt的槽函数和信号机制，进行点击事件的响应，再调用MGraph类中的DFS_NonrecurTraverse()函数和DFS_RecurTraverse()函数进行深度优先遍历，并且用QString连接记录，最后显示即可。

(7) 返回/退出按钮设置

用户点击返回按钮时，将该窗口隐藏，显示要返回的窗口；点击退出按钮时，即可直接退出程序。

实现方式：

Qt的槽函数和信号机制。通过QAction类，QPushButton类实现。

1.3 设计思想

本软件的设计思路就是通过要实现的功能进行按钮，下拉菜单，文本框的位置及功能的设

计。综合运用到核心程序实现的C++代码以及Qt界面设计师提供的界面设计代码，对整个软件进行前端和后台的搭建。

首先，软件的启动界面是通过一个欢迎界面引导，主要包括问题的详细描述，让用户对本软件实现的功能有一个清晰的认知，还有“演示”按钮，进入本软件的核心程序界面，此功能也就是传统的Qt按钮事件——槽函数和clicked()信号进行连接，将启动窗口隐藏(hide)，将目标界面显示(show)。

其次，对于主程序界面，我的设计思想就是首先将按钮(QPushButton)、下拉菜单(QComboBox)、文本显示框(QTextBrowser)的布局及显示界面设计好，功能则利用相应的槽函数的信号连接机制，在某个控件发出信号之后直接触发槽函数，槽函数中实现相应的功能——调用实现核心功能的程序函数。

功能设计思路：

首先，必须得先选择邻接矩阵维度大小及创建的图的种类来进行矩阵的生成，这是创建图的首要条件，在选择矩阵大小和图的种类的时候，选择下拉菜单而不是人工输入，其一是方便设计程序，让程序有更好的辨识度，其二，则是方便用户使用，让选择代替输入在一定程度上省去了不少麻烦。“生成邻接矩阵”按钮的实现，是设置槽函数，当用户发出点击(clicked())信号之后，触发槽函数，在槽函数中调用矩阵生成函数。

其次，生成邻接链表，同样是设置槽函数，在用户发出信号之后，槽函数内调用生成邻接链表的函数，用QString作为返回值，控制在相应位置显示即可。

接着，生成图，同样是槽函数实现的，paintEvent绘图事件是当窗口启动时就进行绘制，而我的想法是通过按钮点击时进行绘制，这样就很容易想到用槽函数机制，只要早槽函数中加repaint()即可每次按钮事件之后都会进行绘制。

其中还有一个需要注意的就是，我设计的图的种类有“有向图”、“无向图”，而Qt::QPainter只提供画直线（只能清晰的表示无向图）功能，不能画出单向箭头（表示有向图），我的解决方法是直线加无底等腰三角形，利用数学知识画出预期的箭头。

最后，两种遍历结果的展示，设置按钮触发槽函数，在槽函数内调用相应的MGraph类函数即可，控制QString返回值，在相应名称的文本框内显示得到的字符串。

为了提高实用性和可操作性，添加“返回起始页”按钮和“退出程序”按钮，让用户操作软件更加灵活。

1.4 逻辑结构与物理结构

(1) 逻辑结构

本软件程序的逻辑结构涉及到图形结构和线性结构。比如通过二维矩阵创建的图和显示的邻接链表，以及栈和队列的创建过程中使用的都是线性结构。

(2) 物理结构

顺序存储结构，在栈和队列的创建过程中，使用的物理存储方式是顺序存储；

链式存储结构，在邻接链表创建的过程中，使用的物理存储方式是链式存储；

1.5 开发平台

(1) 开发平台

计算机型号：Lenovo 小新潮7000

计算机内存：8GB

计算机CPU：Intel Core i7-8550U @1.80GHz @2.00GHz

计算机操作系统类型：64位操作系统，基于x64的处理器

计算机Windows版本：Windows 10 家庭中文版

开发框架：Qt

集成开发环境：Qt 5.12.3

编译器：MinGW 7.3.0 64-bit

(2) 运行环境

在上述集成环境下可调试、运行。

在Qt命令行下转到.exe单独存在的文件夹下，windeployqt xxxx.exe即可发布（release），在任何Windows环境下都可运行。

1.6 系统的运行结果分析说明

(1) 主要功能

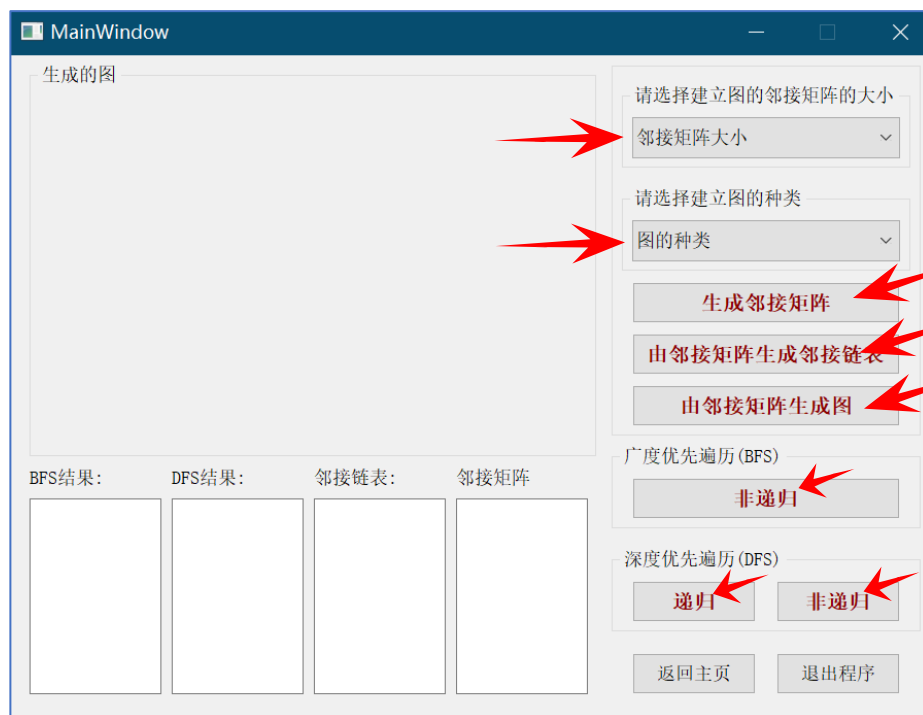
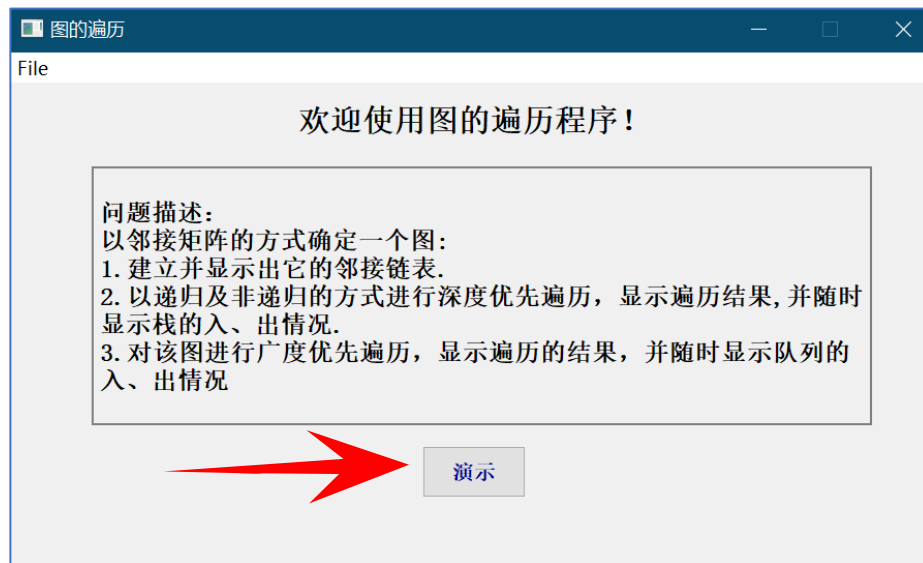
- ① 启动页显示；
- ② 点击进入主应用界面；
- ③ 显示随机生成的指定大小的邻接矩阵、邻接链表、可视化的“图”、深度优先遍历（递归、非递归）结果、广度优先遍历结果；
- ④ 退出\返回按钮；

(2) 软件调试

- ① 稳定性：稳定运行，无崩溃现象；
- ② 容错性：经过各种容错测试，表现符合预期；

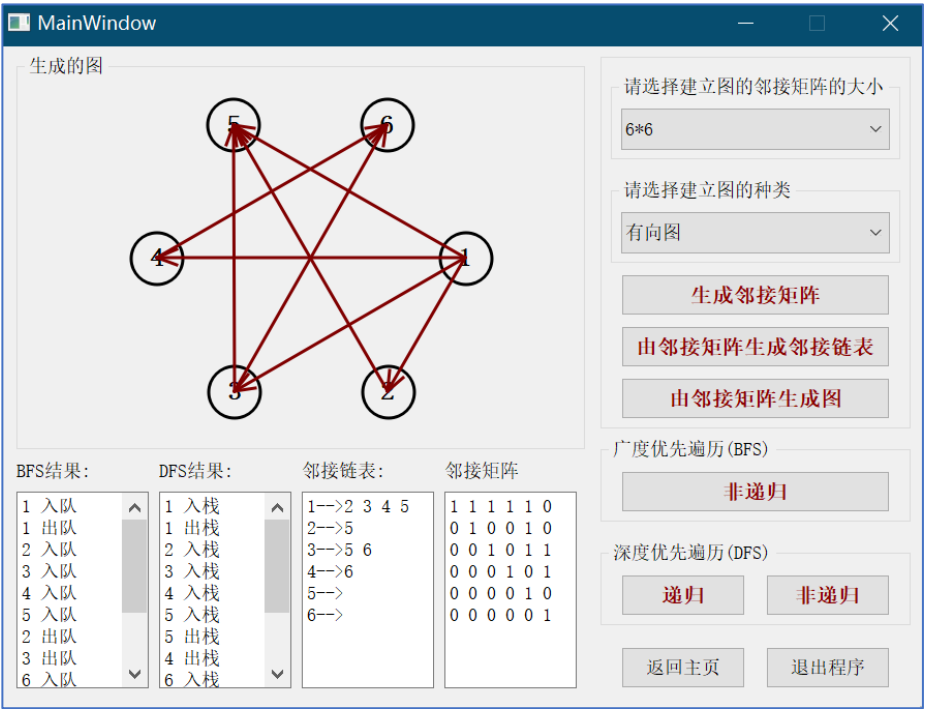
③ 正确性：经过多组数据的测试，结果均表现为正确；

(3) 运行结果

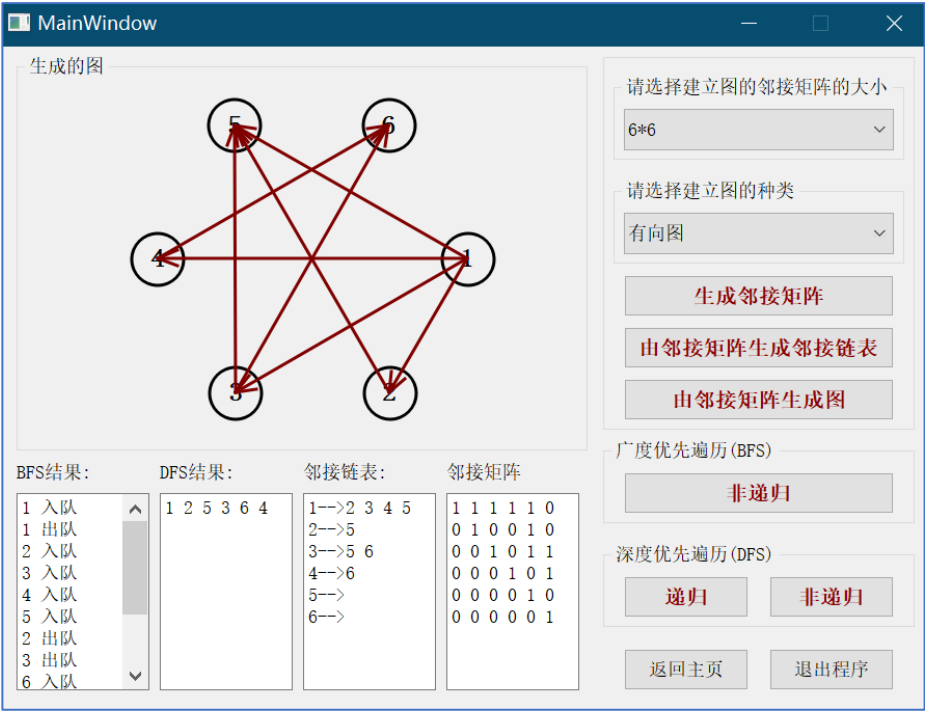


6x6\有向图\邻接矩阵\邻接表\图\bfs——结果如下

dfs非递归:

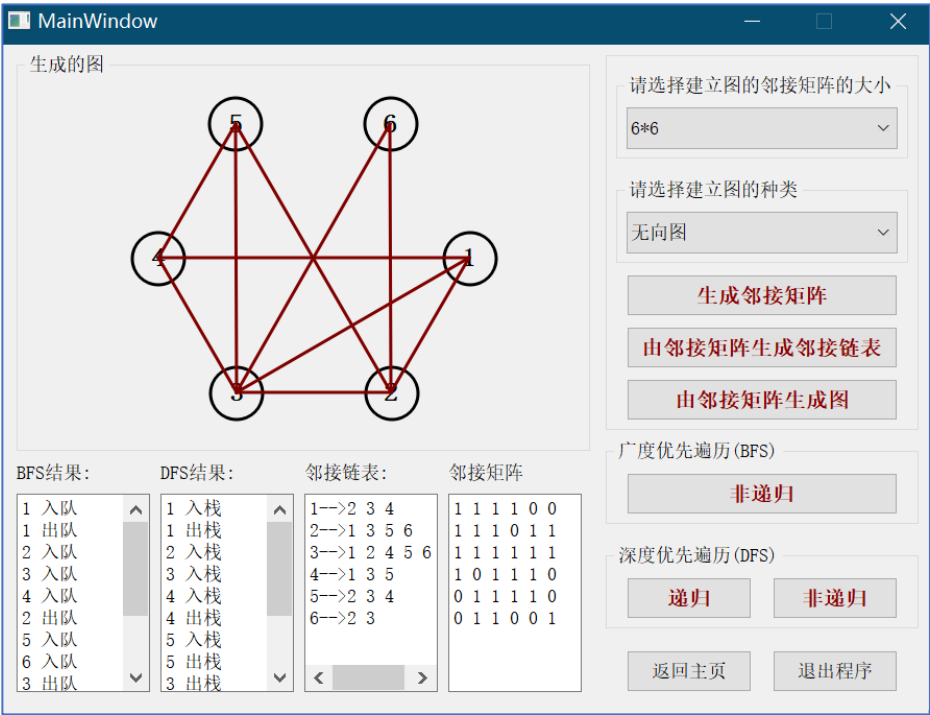


dfs递归:

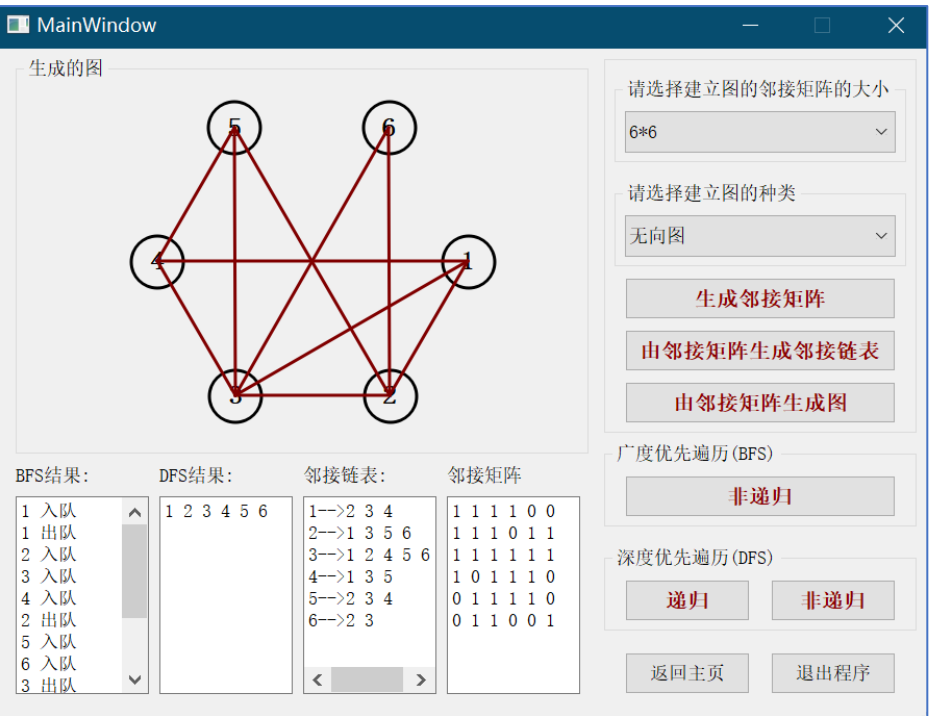


6x6\无向图\邻接矩阵\邻接表\图\bfs——结果如下

dfs非递归:



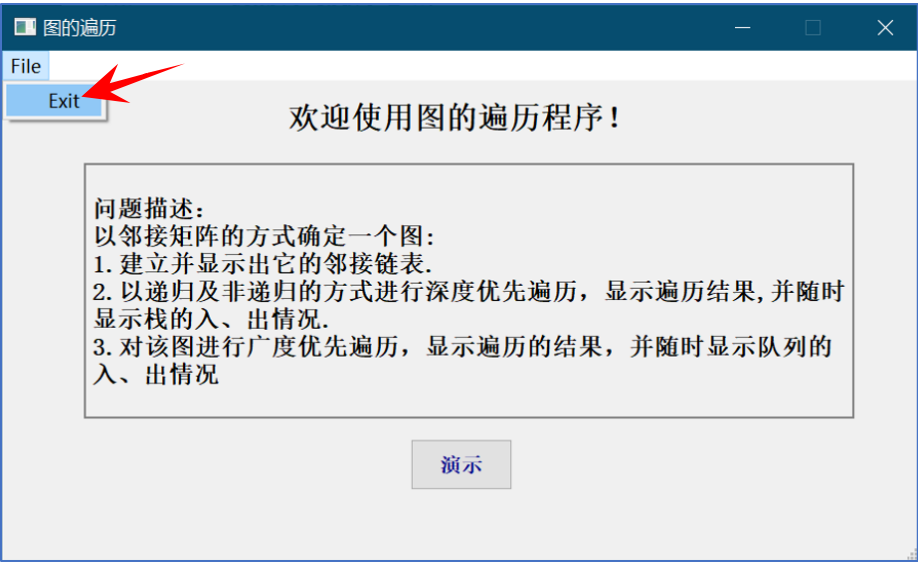
dfs递归:



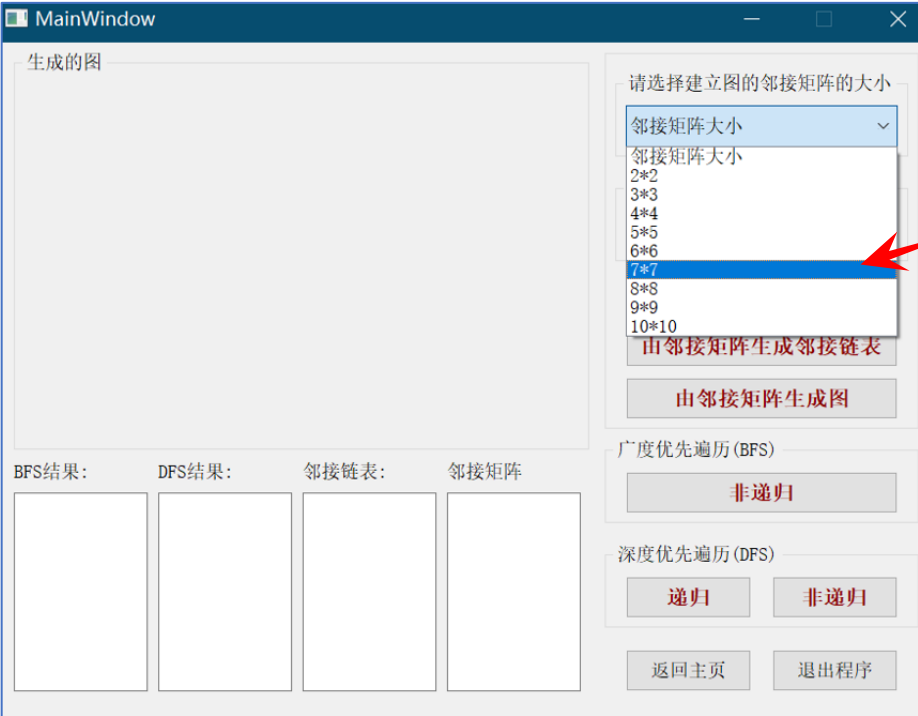
由以上截图可看出，本软件的运行结果符合预期，结果正确。

1.7 操作说明

(1) 启动页菜单栏EXIT退出程序



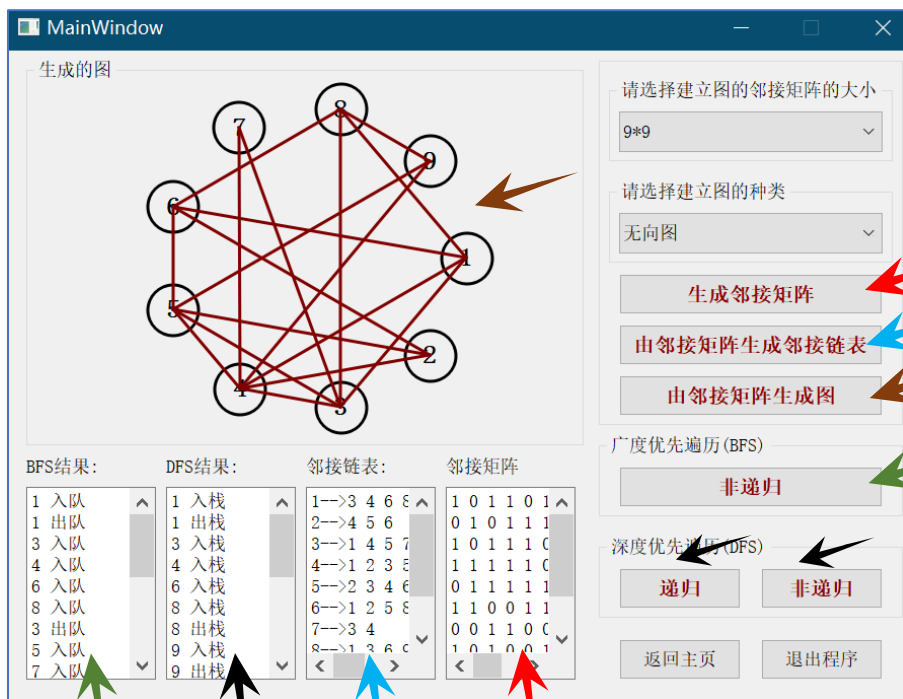
(2) 选择矩阵维度



(3) 选择创建的图的种类

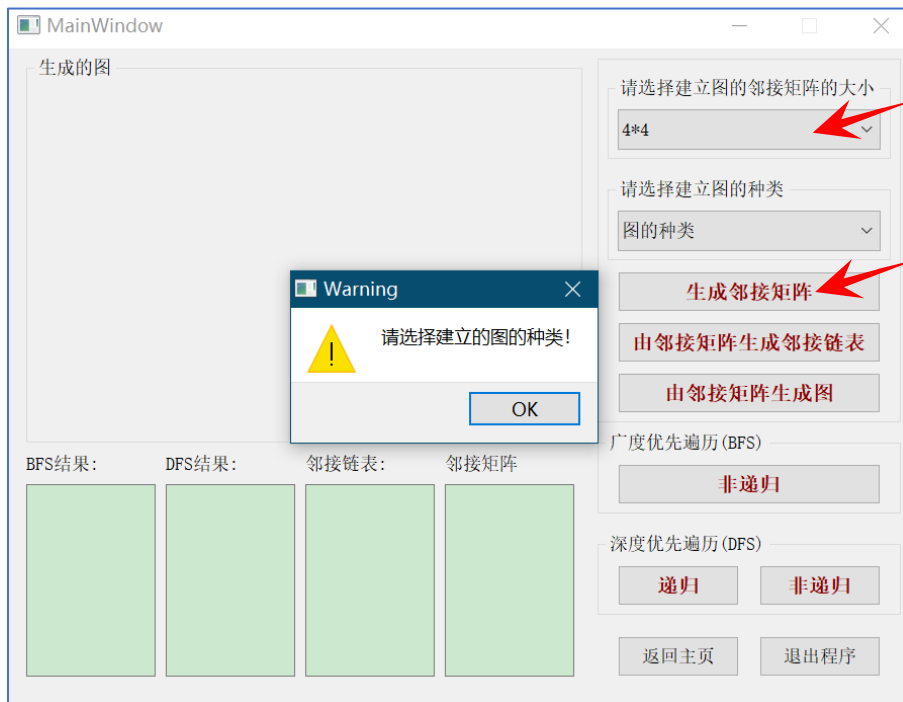


(4) 点击按钮生成相应的内容



(5) 容错性验证

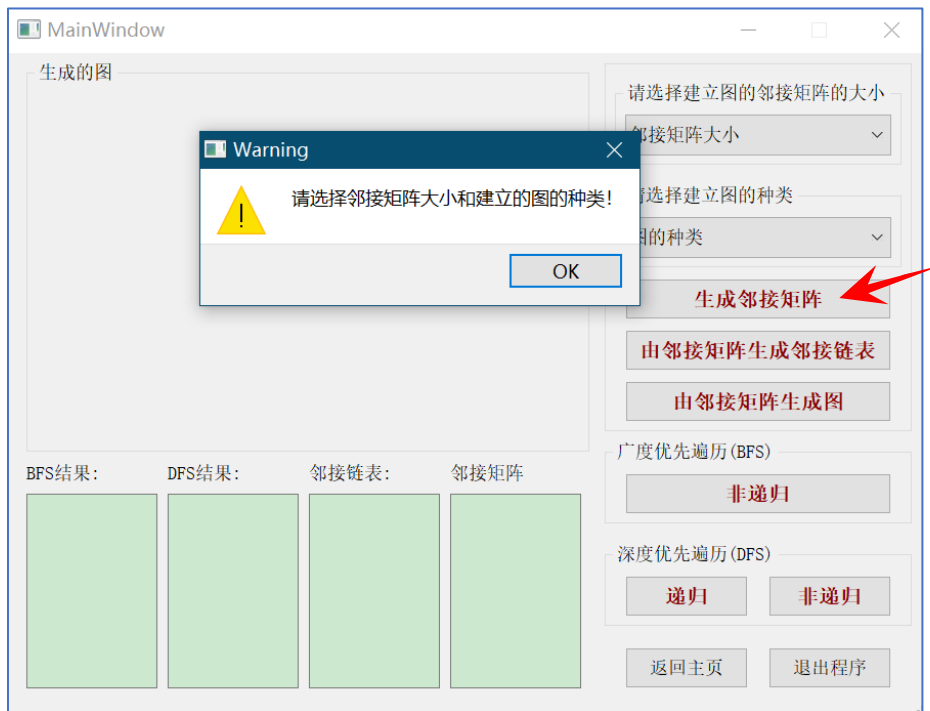
(选择“邻接矩阵维度”之后，没有选择“图的种类”，点击“生成邻接矩阵”按钮)



(选择“图的种类”之后，没有选择“邻接矩阵维度”，点击“生成邻接矩阵”按钮)



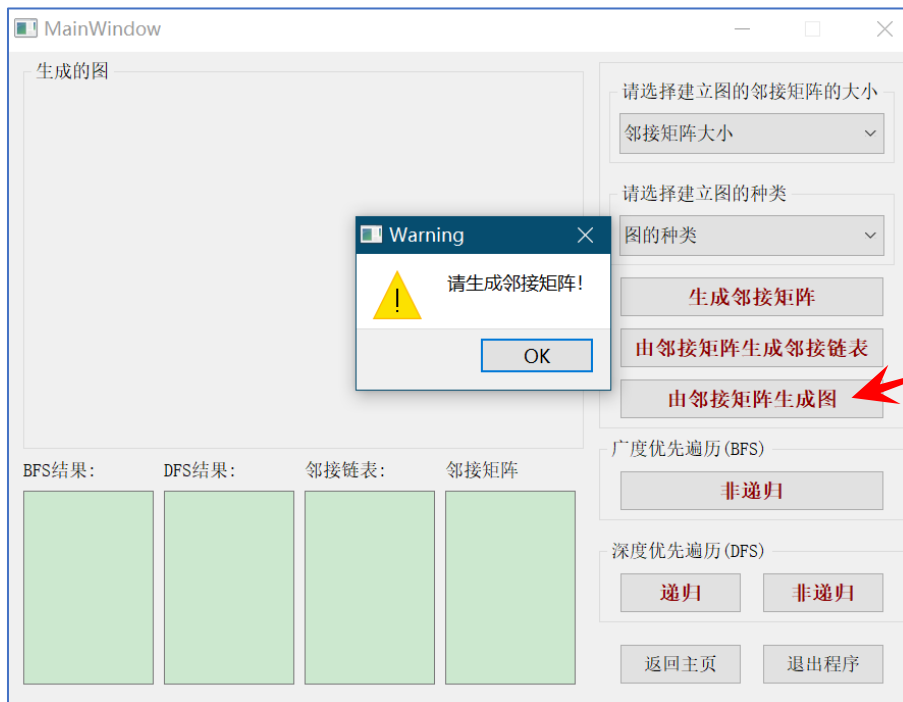
(既没有选择“矩阵维度大小”，又没有选择“图的种类”，点击“生成邻接矩阵”按钮)



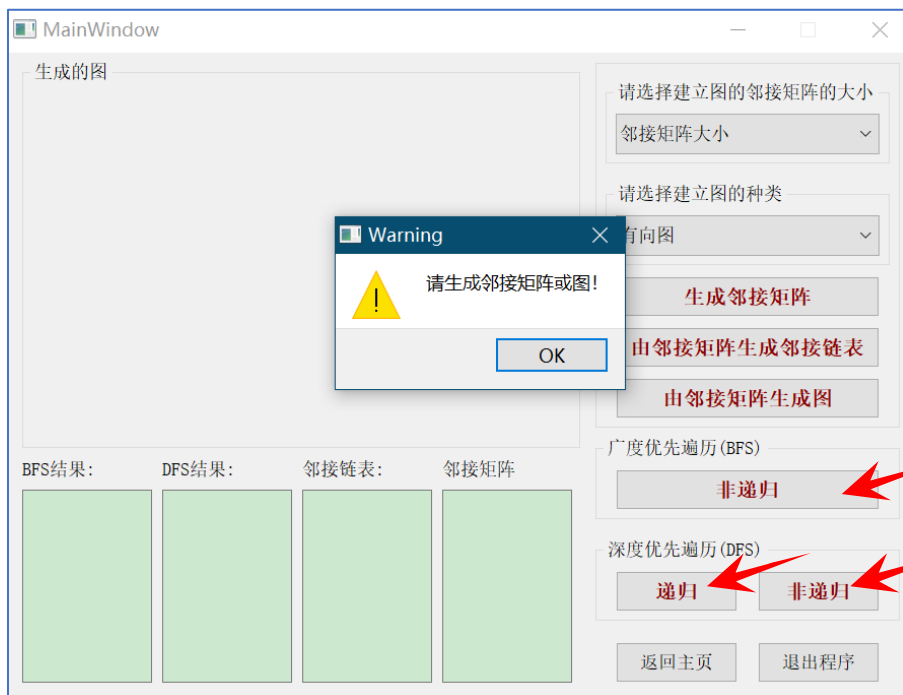
(没有生成邻接矩阵，点击“由邻接矩阵生成邻接链表”按钮)



(没有生成邻接矩阵, 点击“由邻接矩阵生成图”按钮)



(没有生成邻接矩阵或图, 点击“非递归”, “非递归”, “递归”按钮)



第二部分 综合应用设计说明

2.1 题目

在某社会关系网络系统中，一个人属性包括所在地区、就读的各级学校、工作单位等，没一个人有众多好友，并可以根据个人兴趣及社会活动加入到某些群组。现在需要设计一算法，从该社会关系网络中某一个人出发，寻找其可能认识的人。例如根据两个人共同好友数量及所在群组情况，来发现可能认识的人；通过就读学校情况发现可能认识的同学。

- (1) 通过图形化界面，显示某一个人的社会关系网络。
- (2) 寻找某一可能认识的人（不是其好友），并查看这些人与其的关联度（共同好友数）。
- (3) 根据可能认识的关联度对这些人进行排序。

2.2 软件功能

(1) 用户界面菜单栏

File菜单下设置EXIT选项，点击之后退出程序界面。

实现方式：

Qt槽函数和信号连接机制，即点击EXIT选项（产生信号）之后进行触发槽函数，实现相应的动作。

(2) 查看社会关系网络中某节点的详细关系网络

此处用户只需要点击“请选择关系网络中的某节点”即可弹出下拉菜单，选择已经存在于图中的一个节点。点击“查看该节点的关系网络”即可弹出一个窗口——NetworkWindow, 里面包括该结点的所有信息。

实现方式：

通过选择节点菜单QComboBox *selectNode, 构建与之关联的槽函数onCurrentIndexChangedSelectNodeComboBox(), 即当前选择的文本项索引改变时, 调用NetworkWindow的setMGraph和setId函数, 为NetworkWindow类中的mg和id赋值, 并且在画图函数paintEvent(QPaintEvent *)中调用mg->getPersonNode(id), 从而在NetworkWindow窗口显示选择的id节点的相关社会关系网络信息。（此槽函数的触发方式是由QPushButton *nodeNetwork的clicked()信号触发）

（社会关系网络窗口包括，区分好友、可能认识的人、当前节点的图，个人属性（姓名，地区，各级学校，工作单位），兴趣爱好，关联度解析的文本显示窗口）

(3) 添加节点

此处用户只需点击“添加节点”按钮，即可弹出一个添加节点的对话框，里面有各种属性和和单选框以供用户选择兴趣爱好及与其他已生成节点的好友关系，填写完毕所有需要的节点信息之后即可点击“确定按钮”提交所填内容，之后在OperateWindow窗口即可立即显示添加节点之后的图。

实现方式:

通过点击“添加节点”按钮，QPushButton *addNode发出clicked()信号之后触发onClickAddNodeButton()槽函数，调用AddNodeDialog对话框，用户在输入所有信息之后，点击确定退出该对话框之后，再将所有的信息进行记录，即调用AddNodeDialog类里面的setMessage()、setHobby()、setFriend()函数即可将信息记录，之后继续调用getMessa()、getHobby()、getFriend()函数，即可全部得到，从而对OperateWindow里面的MGraph mg进行赋值，用mg.addPersonNode函数向该图添加节点即可，同时还必须动态添加selectNode下拉菜单项内容，将新添加的节点加入其中，再者我们还要在每次加载AddNodeDialog对话框时，调用AddNodeDialog类的setMGraph(MGraph *mg)函数将OperateWindow中的图mg赋值给它，这样每次选择“与其他节点的好友关系”时便可动态添加当前图中的所有节点，从而确定好友关系。

(4) 对当前社会关系网络进行操作

对当前社会关系网络进行操作无非就是“生成节点信息”、“生成邻接链表”、

“生成邻接矩阵”，通过点击不同的按钮，触发不同的槽函数，执行不同的操作，在相应的位置显示即可。

实现方式:

通过点击各自按钮发出clicked()信号，在各自的槽函数中分别调用displayNodesName()、displayAdjMatrix()、displayAdjList()，将当前图的节点信息、当前图代表的邻接矩阵以及当前图代表的邻接链表一一打印到相应的位置

(5) 返回/退出按钮设置

用户点击返回按钮时，将该窗口隐藏，显示要返回的窗口；点击退出按钮时，即可直接退出程序。

实现方式:

Qt的槽函数和信号机制。通过QAction类，QPushButton类实现。

2.3 设计思想

本软件的设计思路首先就是先依照问题需求，将所需要实现的功能按钮和界面布局涉及完整，再将实现的功能代码封装到实体类里面，之后就是利用信号和槽函数机制，点击相应按钮，触发相应的槽函数，调用实体类里面的功能函数，完成相应的功能。

软件本身是由两部分构成，第一部分是启动欢迎界面，第二部分是主程序功能界面。

启动欢迎界面主要包括了程序功能介绍和要求，及进入主程序界面的“演示”按钮。

在点击“演示”按钮之后，进入主程序界面，此时通过点击界面的相应功能按钮，即可实现相应功能。

功能设计思路：

首先，查看节点的“社会关系网络信息”，采用的是下拉菜单列表的形式，方便用户操作，只需根据需求选择相应的节点即可，在此关系网络图中有我初始化的两个朋友关系的节点，因此，在下拉菜单列表中也有初始化的两个节点名称，每当加入一个节点就向其中append它的名称，实现了动态添加。

其次，向当前社会关系网络图中添加节点，采用的是点击按钮，触发槽函数，调用“添加节点”的页面，向其中输入信息，选择兴趣爱好，选择与“当前图中的节点”的好友关系，其中“与其他好友节点关系”中的好友节点多选框则需要动态添加（我的思路是，每次在操作页面OperateWindow中新建一个AddNodeDialog，这样就可以每次将操作窗口的mg传给添加节点页面的mg，从而对该图节点实现遍历，这样每次新添加的节点都会在加载添加节点窗口之前被赋值，加载之后就会显示）

接着，是生成当前图的所有信息，在此我设计三个按钮对图进行操作，分别发信号触发槽函数，调用相应的实体类功能函数，将返回的QString类型字符串显示在相应名称的QTextBrowser文本框里面，功能得以实现。

最后，为了提高程序的实用性和可操作性，我在每个页面的最后都添加了“返回”和“退出程序”按钮，这些当然也都是通过信号-槽函数机制实现的。

2.4 逻辑结构与物理结构

(1) 逻辑结构

本软件程序的逻辑结构涉及到图形结构和线性结构。比如通过二维矩阵创建的图和显示的邻接链表，使用的都是线性结构。

(2) 物理结构

顺序存储结构，在二维矩阵和数组的创建过程中使用的物理存储方式是顺序存储方式；
链式存储结构，在邻接链表创建的过程中，使用的物理存储方式是链式存储；

2.5 开发平台

(1) 开发平台

计算机型号：Lenovo 小新潮7000

计算机内存：8GB

计算机CPU：Intel Core i7-8550U @1.80GHz @2.00GHz

计算机操作系统类型：64位操作系统，基于x64的处理器

计算机Windows版本：Windows 10 家庭中文版

开发框架：Qt

集成开发环境：Qt 5.12.3

编译器：MinGW 7.3.0 64-bit

(6) 运行环境

在上述集成环境下可调试、运行。

在Qt命令行下转到.exe单独存在的文件夹下，windeployqt xxxx.exe即可发布（release），在任何Windows环境下都可运行。

2.6 系统的运行结果分析说明

(1) 主要功能

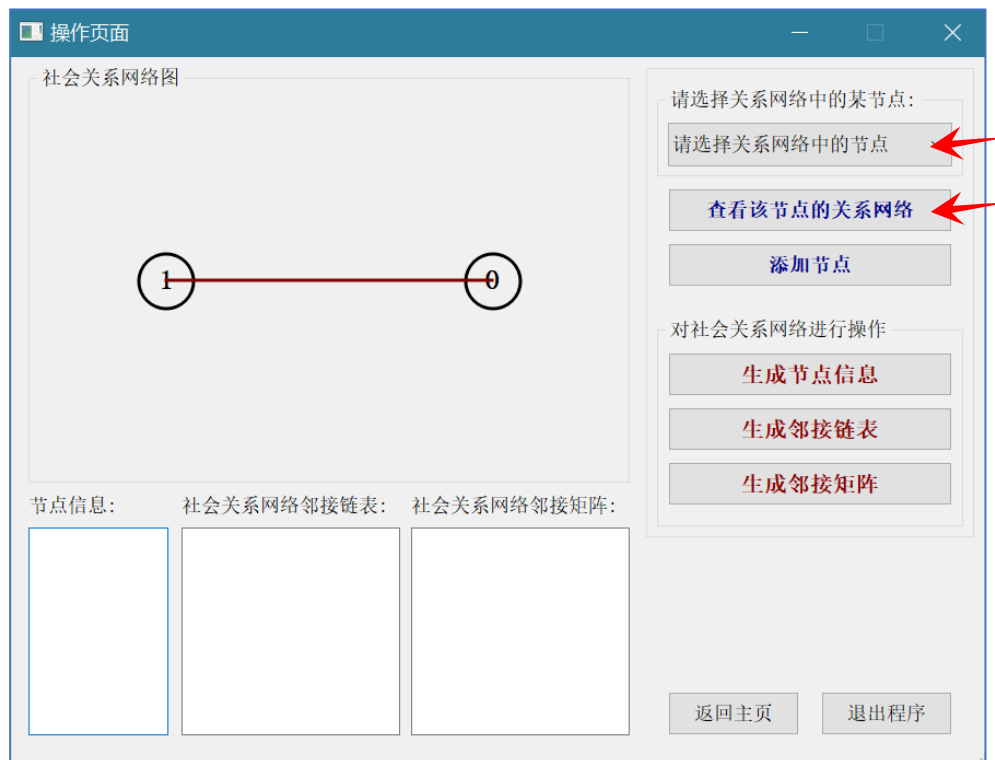
- ① 显示启动窗口；
- ② 进入主程序界面；
- ③ 选择图中的节点，查看该节点的详细社会关系网络，向当前关系网络中添加新的节点，显示添加新节点的窗口，生成当前关系网络的节点具体信息，邻接矩阵，邻接链表。
- ④ 在每个窗口添加返回和退出程序按钮。

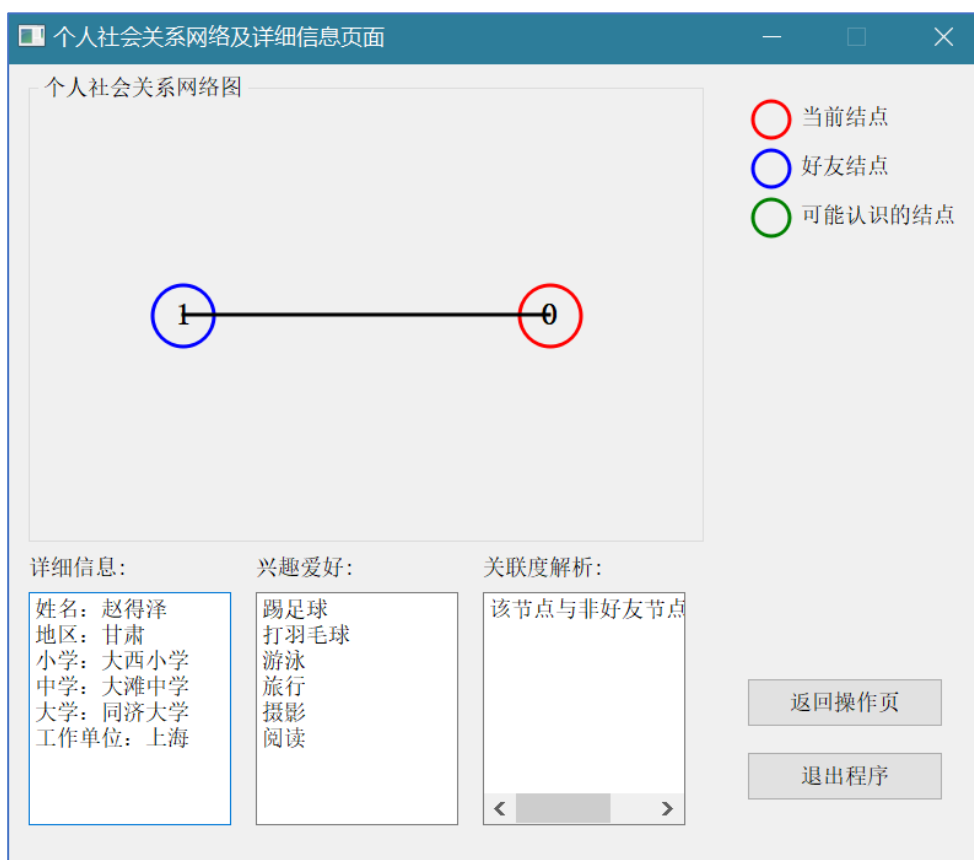
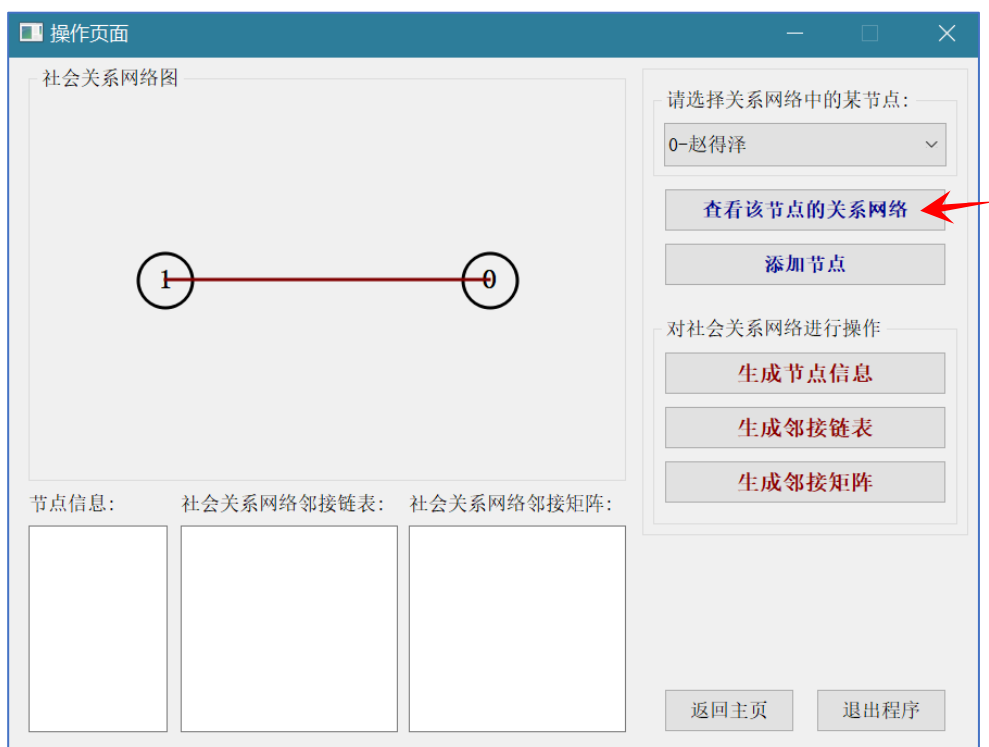
(2) 调试运行

- ① 稳定性：软件无崩溃情况，稳定性正常；
- ② 容错性：经多方面考虑和测试，软件容错性合乎逻辑且正常。

③ 正确性：径控制台输出和自己对比，运行结果符合预期。

(3) 运行结果





操作页面

社会关系网络图

节点信息:

社会关系网络邻接链表:

社会关系网络邻接矩阵:

请选择关系网络中的某节点:

请选择关系网络中的节点

查看该节点的关系网络

添加节点

对社会关系网络进行操作

生成节点信息

生成邻接链表

生成邻接矩阵

返回主页

退出程序



添加节点页面

姓名

李鹏飞

地区

河南

小学

东镇小学

中学

民勤五中

大学

兰州大学

工作单位

兰大二院

兴趣爱好:

☒ 打篮球

☒ 踢足球

☐ 打乒乓球

☐ 打羽毛球

☐ 游泳

☒ 旅行

☐ 摄影

☒ 阅读

☐ 编程

☒ 看电影

与其他节点的好友关系:

☒ 0-赵得泽

☒ 1-汤海迪

确定

取消



操作页面

社会关系网络图

节点信息:

社会关系网络邻接链表:

社会关系网络邻接矩阵:

请选择关系网络中的某节点:

请选择关系网络中的节点

请选择关系网络中的节点

0-赵得泽

1-汤海迪

2-李鹏飞

添加节点

对社会关系网络进行操作

生成节点信息

生成邻接链表

生成邻接矩阵

返回主页

退出程序



添加节点页面

姓名

地区

小学

中学

大学

工作单位

兴趣爱好:

☐ 打篮球

☐ 踢足球

☐ 打乒乓球

☐ 打羽毛球

☐ 游泳

☐ 旅行

☐ 摄影

☐ 阅读

☐ 编程

☐ 看电影

与其他节点的好友关系:

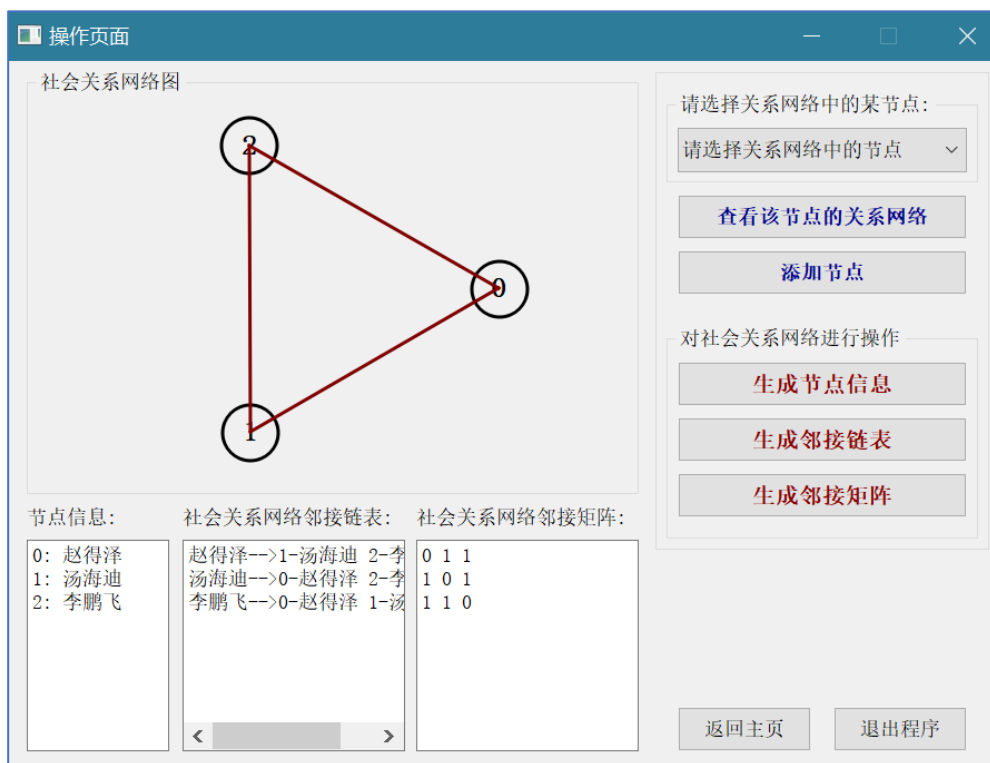
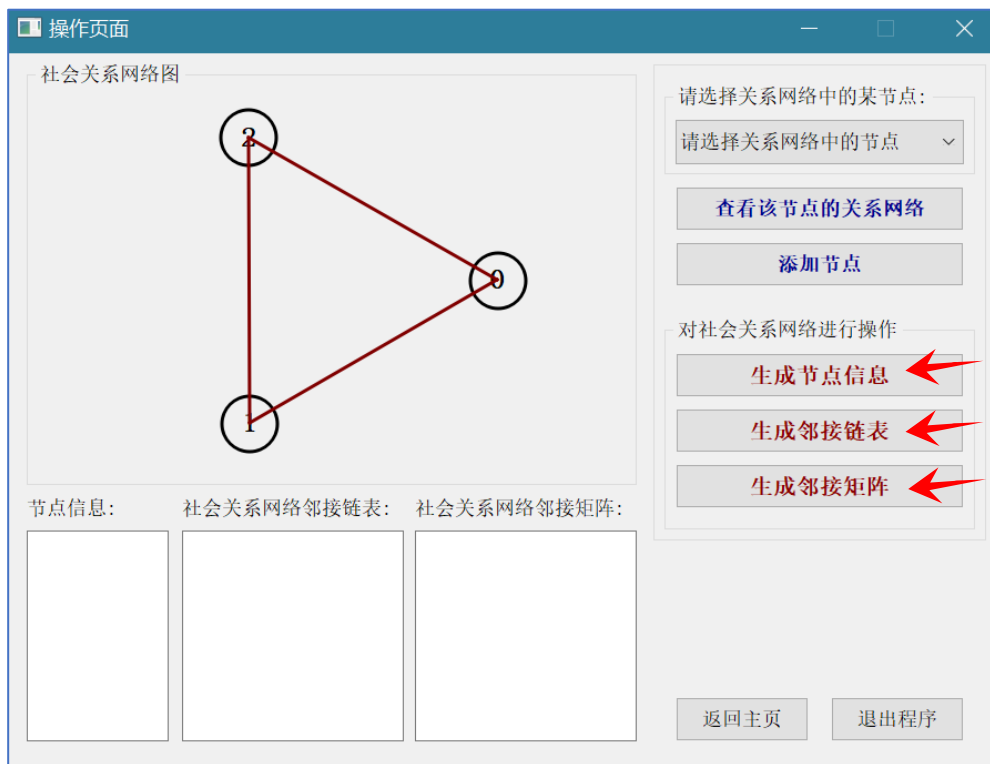
☐ 0-赵得泽

☐ 1-汤海迪

☐ 2-李鹏飞

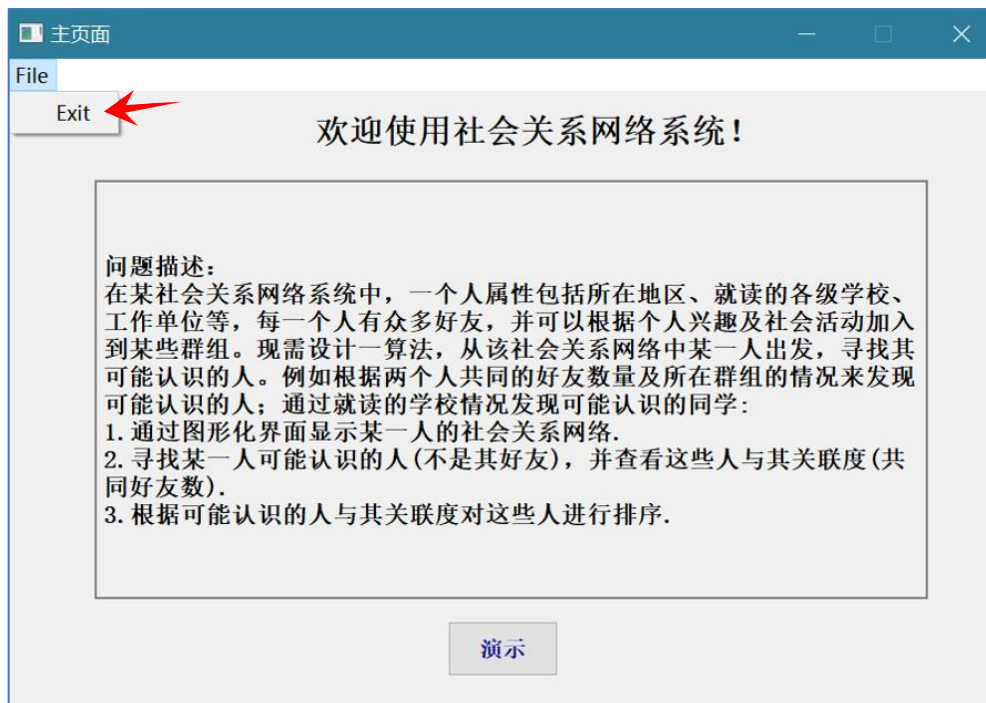
确定

取消

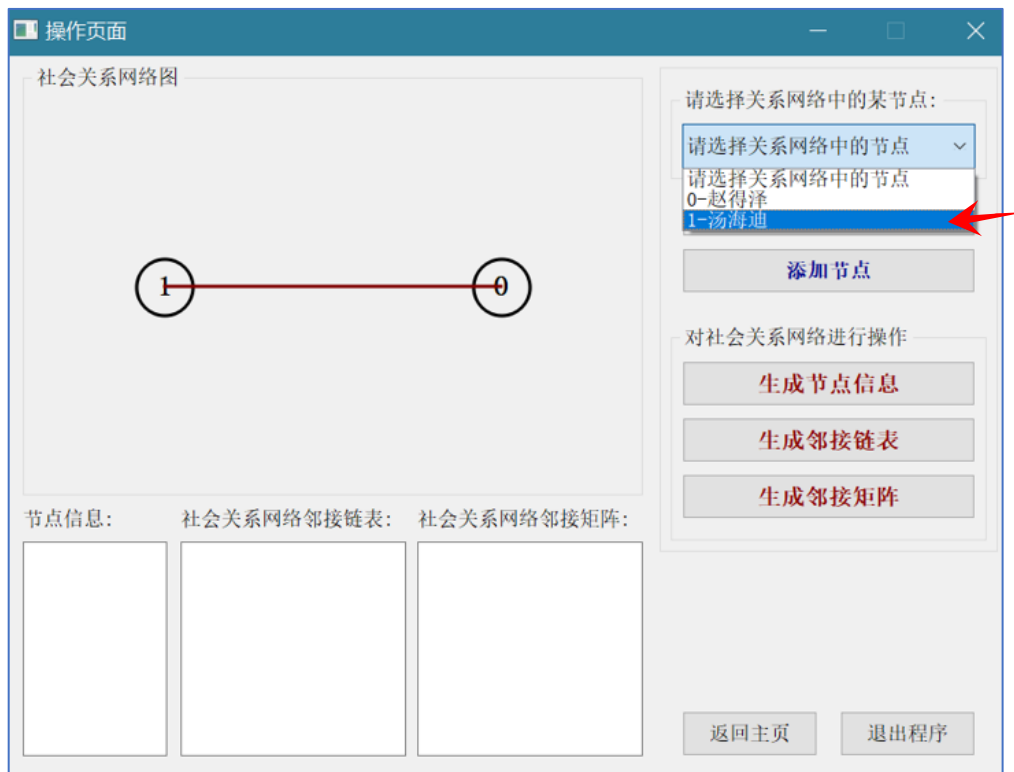


2.7 操作说明

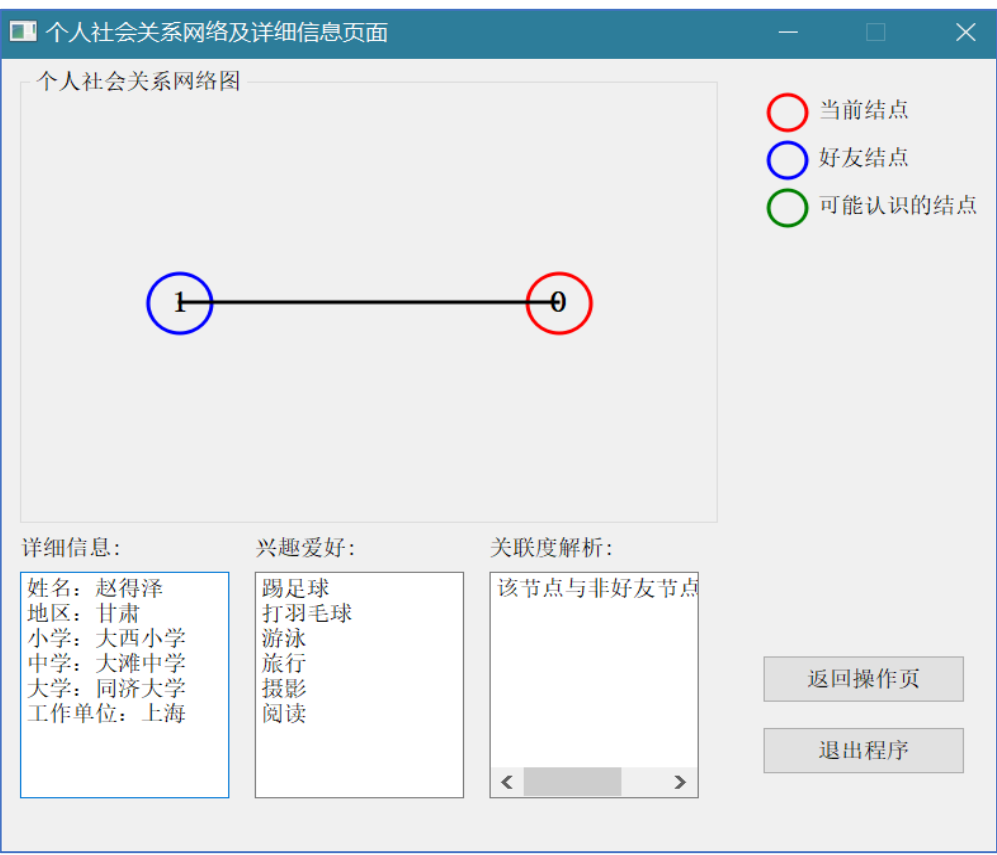
(1) 菜单栏的EXIT退出程序功能



(2) 选择要查看的当前关系网络某节点详细信息

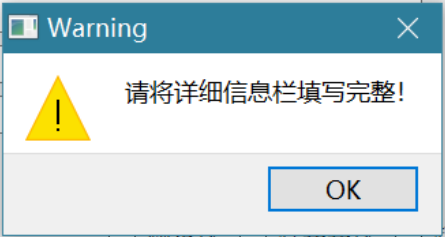


(3) 点击“查看该节点的关系网络”



(4) 点击“添加节点”

(5) 在没填写详细信息时点击“确定”



添加节点页面

姓名:

地区:

小学:

中学:

大学:

工作单位:

兴趣爱好:

☐ 打篮球 ☐ 踢足球 ☐ 打乒乓球 ☐ 打羽毛球 ☐ 游泳

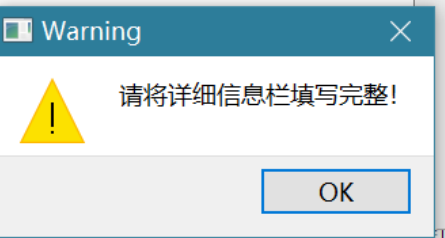
☐ 旅行 ☐ 摄影 ☐ 阅读 ☐ 编程 ☐ 看电影

与其他节点的好友关系:

☐ 0-赵得泽 ☐ 1-汤海迪

确定 取消

(6) 在填写部分详细信息时点击“确定”



添加节点页面

姓名:

地区:

小学:

中学:

大学:

工作单位:

兴趣爱好:

☒ 打篮球 ☒ 踢足球 ☒ 打乒乓球 ☒ 打羽毛球 ☐ 游泳

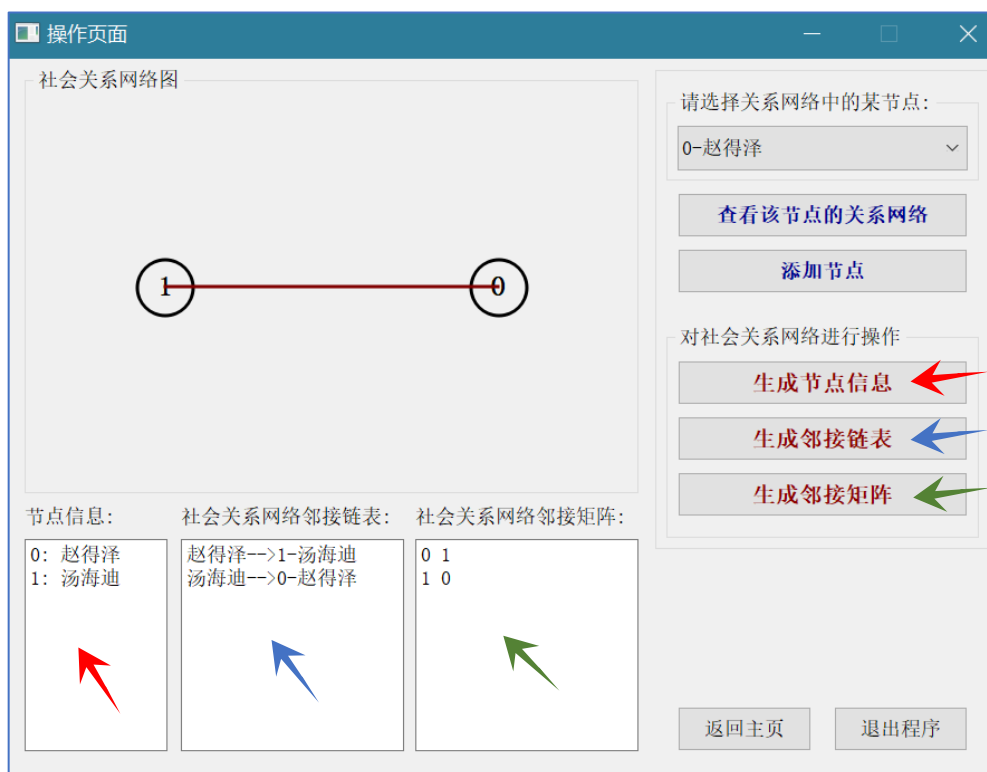
☒ 旅行 ☐ 摄影 ☐ 阅读 ☒ 编程 ☐ 看电影

与其他节点的好友关系:

☒ 0-赵得泽 ☐ 1-汤海迪

确定 取消

(7) 点击“对社会关系网络进行操作”三个按钮



(8) 其他容错性验证

(没有选择要查看的关系网络的节点，直接点击“查看该节点的关系网络”)



(节点数目限制, 当社会关系网络节点数目达到10个之后, 不能添加节点)

操作页面

社会关系网络图

Warning

! 结点数目已达上限! 最多不能超过10个!

OK

节点信息:

2: a
3: b
4: c
5: d
6: e
7: f
8: g
9: h

社会关系网络邻链表:

赵得泽-->1-汤海迪 2
汤海迪-->0-赵得泽 4
a-->0-赵得泽 3-b 4-
b-->0-赵得泽 2-a
c-->1-汤海迪 2-a 5-
d-->1-汤海迪 4-c 7-
e-->0-赵得泽 8-g
f-->1-汤海迪 5-d 0-

社会关系网络邻接矩阵:

0	1	1	1	0	0	1	0	1	0
1	0	0	0	1	1	0	1	0	0
1	0	0	1	1	0	0	0	0	1
1	0	1	0	0	0	0	0	0	0
0	1	1	0	0	1	0	0	0	0
0	1	0	0	1	0	0	1	0	0
1	0	0	0	0	0	0	1	0	0
0	1	0	0	0	1	0	0	0	1

请选择关系网络中的某节点:

6-e

查看该节点的关系网络

添加节点

生成节点信息

生成邻接链表

生成邻接矩阵

返回主页

退出程序

第三部分 实践总结

3.1 所做的工作

本次课程设计我所完成的内容是：图的遍历和社会关系网络系统。

图的遍历，本题目要求根据邻接矩阵，显示图的邻接链表和遍历结果，此题我为了节省用户输入邻接矩阵的时间，直接将选择了用下拉菜单选择矩阵大小和图的种类来创建图，矩阵生成的思想是随机生成矩阵的上三角，对角线上的值置1，再根据图的种类来生成整个邻接矩阵，之后就是根据邻接矩阵生成邻接链表，生成可视化的图，通过BFS进行非递归遍历图，并且输出出队入队的情况及结果序列，通过DFS进行递归遍历图，生成遍历图的结果序列，还有非递归遍历图，生成结果序列和出栈入栈的情况。

不足之处就是没有将入、出队和入、出栈进行图形可视化处理，看起来比较生硬。

社会关系网络系统，本题目要求找出一个人的关系网络，并且可以随意添加节点，找出一个人可能认识的人，并且排序。此题目我首先初始化了两个节点作为可视化的图，在此基础上可以直接添加节点，并且实时显示添加新节点后构成的新图，另外，还可以输出当前图的节点信息、邻接链表、邻接矩阵。

不足之处依然还是邻接链表没有可视化，看起来比较生硬，还有就是当查看某个人的社会关系网络的时候，要用下拉菜单实现，用户操作起来可能不太方便，做成直接点击的圆形按钮似乎更能体现它的方便和简洁，还有一点就是添加节点的时候可能需要用户输入的东西还是有些多，尽量做成选择的下拉菜单会更方便一些。

3.2 总结与收获

本次我选的两个题目主要运用的知识是数据结构里面的图形结构，还用到少量的栈和队列的知识。为了能够更熟练的运用知识，将原来的知识又温习了一遍，并且又模拟了一次代码。最后根据题目需要将这几个数据结构进行整合。本次课设最主要的目的是考察我们知识的综合运用能力，准确的来说是将它做成一个小软件，这就不仅仅限于代码层面了，更多的还要有注释，文档说明等等。

自我学习能力方面，首先就是关于新软件Qt的运用，在此之前，我有关于Qt使用的一些经验，但还是不能够将本次的课设圆满完成。所以在网络上找到了一些关于Qt操作的知识，查阅了大量的资料，并且对GUI编程有了更深刻的理解，同时对自身的成长也有着不可忽视的作用，Qt主要是将所有的累都进行了封装，这样既方便了调用，又提高了数据的安全性和封闭

性，作为专用于C++的GUI编程框架，它很完美地体现了C++面向对象编程的思想，同时我对C++也有了更深一步的认识。在一步步查阅资料，理解每个功能如何实现的时候，我也逐渐理解了Qt框架中的信号和槽机制，所有的函数都是通过用户对某个控件的控制使其发出信号，然后将相应的槽函数与控件绑定，由信号去触发相应的槽函数，被调用而实现它的功能的。

交互界面的设计和布局是我这次课设学到的另一个十分有价值的知识，Qt里面很多对于界面布局的内置函数我也有了全新的认识，除了直接在Qt Designer进行拖拽设计之外，我也掌握了通过控制条件、变量去动态设计一个控件，并且通过不同的响应添加控件，使得程序更富有逻辑性和条理性。

这次设计是一次全新的、将C++从console控制台窗口移植向面向用户的可视化界面的体验。在不断增长我见识的同时，也让我能够对C++有更加全面深刻的了解，对数据结构有了更独到的认识，debug的过程也正是增强自身能力的过程，此次设计过程中我遇到过很多困难，包括程序出现的各种各样的bug，但作为计算机系的学生，首先就必须得有debug的能力和耐心，这对我也是一种考验和磨练，总之，本次课设带给我很多有形和无形的知识，我也从中颇获裨益。

第四部分 参考文献

- [1] 严蔚敏, 吴伟民. 数据结构(C语言版), 清华大学计算机系列教材. 北京: 清华大学出版社, 2018. 6
- [2] 一念之间、, Qt学习之路14—布局管理器(QGridLayout),
https://blog.csdn.net/tqs_1220/article/details/81842878
- [3] 马孔多的黑夜, Qt值QgroupButton的使用,
<https://blog.csdn.net/u013782830/article/details/53392072>
- [4] Fairystepwgl, Qt中关键字讲解, (emit, signal, slot)
<https://www.cnblogs.com/felix-wang/p/6212197.html>
- [5] 未狂, 【Qt】窗体间传递数据(跨控件跨类), 三种情况与处理方法,
<https://blog.csdn.net/shihoongbo/article/details/48681979>
- [6] 杨凯楠, QT Qdialog的对话框模式以及其关闭,
<https://cnblogs.com/felix-wang/p/6212197.html>
- [7] 一去、二三里, Qt 之水平/垂直布局(QBoxLayout、QHBoxLayout、QVBoxLayout)
<https://blog.csdn.net/liang19890820/article/details/51537246>