《Java语言及网络编程》作业三

学号	姓名	班级
05191643	许万鹏	信息安全19-01班

1 第一题

1.1 题目

- 用泛型List管理学生信息，学生对象信息为：姓名、学号、年龄、专业
- 对学生列表完成：添加、删除、查找、全部显示操作。

1.2 答案

1.2.1 项目结构树

```
D:.
|   ...
└─src
    └─edu
        └─wanpengxu
            └─homework3
                └─first
                        Student.java
                        StudentAdministrator.java
                        Test.java
```

1.2.2 代码

Student.java

```
package edu.wanpengxu.homework3.first;

public class Student {
    private String name;
    private String studentID;
    private int age;
    private String major;

    public Student(String name, String studentID, int age, String major) {
        this.name = name;
        this.studentID = studentID;
        this.age = age;
        this.major = major;
    }

    public String getName() {
        return name;
    }
```

```
    public String getStudentID() {
        return studentID;
    }

    public int getAge() {
        return age;
    }

    public String getMajor() {
        return major;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setStudentID(String studentID) {
        this.studentID = studentID;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public void setMajor(String major) {
        this.major = major;
    }

    public void display() {
        System.out.println("name:" + name);
        System.out.println("studentID:" + studentID);
        System.out.println("age:" + age);
        System.out.println("major:" + major);
        System.out.println();
    }
}
```

StudentAdministrator.java

```
package edu.wanpengxu.homework3.first;

import java.util.ArrayList;
import java.util.List;

public class StudentAdministrator {
    private static final List<Student> database = new ArrayList<>();  // database变量始终引用
数组
```

```
    public boolean add(Student student) {
        if(student.getStudentID().length()!=8){
            return false;
        }
        for(Student studentI:database){
            if(studentI.getStudentID().equals(student.getStudentID())){
                return false;
            }
        }
        return database.add(student);
    }

    public List<Student> find(String name) {
        List<Student> foundStudent = new ArrayList<>();
        for (Student student : database) {
            if (student.getName().equals(name)) {
                foundStudent.add(student);
            }
        }
        return foundStudent;
    }

    public boolean delete(String studentID) {
        return database.removeIf(student -> student.getStudentID().equals(studentID));
    }

    public void showAll() {
        for (Student student : database) {
            student.display();
        }
    }
}
```

Test.java

```
package edu.wanpengxu.homework3.first;

import java.io.Console;
import java.util.*;

public class Test {
    public static void main(String[] args) {
        HashMap<String, String> administrators = new HashMap<>();
        StudentAdministrator studentAdministrator = null;
        System.out.println("Welcome to undergraduate management system!");
        try (Scanner scanner = new Scanner(System.in)) {
            boolean logged = false;
            while (true) {
```

```
                System.out.println("Please select: [C]Creat
Administrator\t[L]Login\t[A]Add\t[D]Delete\t[F]Find\t[S]Show All\t[E]Exit");
                String type = scanner.next();
                if (type.equalsIgnoreCase("C")) {
                    System.out.println("Please input your username:");
                    String username = scanner.next();
                    System.out.println("Please input your password:");
                    String password = scanner.next();  // 不隐藏输入
//                    Console con = System.console();
//                    String password = new String(con.readPassword());  // 隐藏输入，在IDE中
不可行，因为控制台被重定向了
                    if (administrators.get(username) == null) {
                        administrators.put(username, password);
                        System.out.println("Creat succeeded!");
                    } else {
                        System.out.println("Creat failed!");
                    }
                } else if (type.equalsIgnoreCase("L")) {
                    System.out.println("Please input your username:");
                    String username = scanner.next();
                    System.out.println("Please input your password:");
                    String password = scanner.next();
//                    Console con = System.console();
//                    String password = new String(con.readPassword());
                    if (administrators.get(username) != null &&
administrators.get(username).equals(password)) {
                        System.out.println("Login succeeded!");
                        logged = true;
                        studentAdministrator = new StudentAdministrator();
                    } else {
                        System.out.println("Login failed!");
                    }
                } else if (type.equalsIgnoreCase("E")) {
                    System.out.println("Exit succeeded!");
                    break;
                } else {
                    if (logged) {
                        if (type.equalsIgnoreCase("A")) {
                            System.out.println("Please input student's name:");
                            String name = scanner.next();
                            System.out.println("Please input 8-digit student's studentID:");
                            String studentID = scanner.next();
                            System.out.println("Please input student's age:");
                            int age;
                            try {
                                age = scanner.nextInt();
                            } catch (InputMismatchException ex) {
                                System.out.println("Invalid input!");
```

```
                                scanner.next();  // 读入并丢弃异常的输入
                                continue;
                            }
                            System.out.println("Please input student's major:");
                            String major = scanner.next();
                            if (studentAdministrator.add(new Student(name, studentID, age,
major))) {
                                System.out.println("Added successfully!");
                            } else {
                                System.out.println("Added failed!");
                            }
                        } else if (type.equalsIgnoreCase("D")) {
                            System.out.println("Please input 8-digit student's studentID:");
                            String studentID = scanner.next();
                            if (studentAdministrator.delete(studentID)) {
                                System.out.println("Delete successfully!");
                            } else {
                                System.out.println("Delete failed!");
                            }
                        } else if (type.equalsIgnoreCase("F")) {
                            System.out.println("Please input student's name:");
                            String name = scanner.next();
                            List<Student> found = studentAdministrator.find(name);
                            System.out.println("Find " + found.size() + " students!");
                            for (Student student : found) {
                                student.display();
                            }
                        } else if (type.equalsIgnoreCase("S")) {
                            studentAdministrator.showAll();
                        }
                    } else {
                        System.out.println("Please login first!");
                    }
                }
            }
        }
    }
}
```

1.2.3 运行截图

```
 D:\Codefield\CODE_Java\IdeaProjects\src\edu\wanpengxu\homework3\first
λ javac *.java

 D:\Codefield\CODE_Java\IdeaProjects\src\edu\wanpengxu\homework3\first
λ cd D:\Codefield\CODE_Java\IdeaProjects\src

 D:\Codefield\CODE_Java\IdeaProjects\src
λ java edu.wanpengxu.homework3.first.Test
Welcome to undergraduate management system!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
a
Please login first!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
C
Please input your username:
admin
Please input your password:

Creat succeeded!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
L
Please input your username:
root
Please input your password:

Login failed!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
L
Please input your username:
admin
Please input your password:

Login succeeded!
```

编译Java包

不区分大小写

访问控制

密码隐藏

```
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
A
Please input student's name:
许万鹏
Please input 8-digit student's studentID:
05191643
Please input student's age:
20
Please input student's major:
信息安全
Added successfully!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
A
Please input student's name:
许万鹏
Please input 8-digit student's studentID:
06211234
Please input student's age:
18
Please input student's major:
经管类
Added successfully!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
A
Please input student's name:
冯诺依曼
Please input 8-digit student's studentID:
08030001
Please input student's age:
54
Please input student's major:
计算机科学与技术
Added successfully!
```

添加操作

```
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
A
Please input student's name:
吴恩达
Please input 8-digit student's studentID:
08030002
Please input student's age:
四十五岁
Invalid input!
```

```
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
S                                                                                 全部显示操作
name:许万鹏
studentID:05191643
age:20
major:信息安全

name:许万鹏
studentID:06211234
age:18
major:经管类

name:冯诺依曼
studentID:08030001
age:54
major:计算机科学与技术

Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
F                                                                  查找操作（按名查找）
Please input student's name:
许万鹏
Find 2 students!
name:许万鹏
studentID:05191643
age:20
major:信息安全

name:许万鹏
studentID:06211234
age:18
major:经管类
```

```
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
D                                                                  删除操作
Please input 8-digit student's studentID:
06211234
Delete successfully!
Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
S
name:许万鹏
studentID:05191643
age:20
major:信息安全

name:冯诺依曼
studentID:08030001
age:54
major:计算机科学与技术

Please select: [C]Creat Administrator   [L]Login        [A]Add  [D]Delete       [F]Find [S]Show All     [E]Exit
E                                                                                                   退出操作
Exit succeeded!
```

1.3 分析

一道比较简单的模拟题，大二时我们在程序设计综合实践课上做过更加复杂的例子。

首先我认为需要有学生类Student

还需要一个学生管理员类StudentAdministrator用于管理学生

最后在主类里调用学生管理员对象去管理学生对象

要注意的是这里涉及到了数据库的知识，学生表（在这里是`List<Student>`）的主键是学号，所以添加、删除这些操作都需要使用学号完成，同时考虑到非主键会有重复，所以查找操作使用了姓名，当然这种写法只适用于小型系统，如学院、学校的数据库系统，如果同名人数多的系统，如某省的户籍系统，还需要加入其他限定条件。

在基本要求外做了一个比较简单的访问控制，并没有保存它使其持久化。

注意到控制台程序的每个选项间距都不同，因为制表符`\t`用空格补全其左边字符串，使其长度到8的整数倍（正好是8的倍数也补，不然体现不出按制表键）。

2 第二题

2.1 题目

一元二次方程类:

1. 输入一元二次方程的三个系数，确定这个方程式。

   标明一个一元二次方程的属性。

   - 三个系数

     a, b, c

   - 两个根(最多)(而且可能是虚根所以用统一的形式表示)

     x_1,y_1, x_2,y_2

   - 决定根的特征的

     $\delta = b^2 - 4ac$

   如果是一个一元二次方程需要的操作（注意方法的权限）：

   - 获取三个系数,确立一元二次方程.
   - 求$\delta = b^2 - 4ac$
   - 求根(可能是实根,也可能是虚根)
2. 如果二次系数a为0 ，主动抛出异常，因为这不满足一元二次方程的条件。

   ```
   public class AWrongException extends Exception
   ```

3. 把生成的一元二次方程对象（包含根的值）序列化并保存在文件`file1.data`中。

   - 根求出之后，再把对象序列化到文件file1.data中。

要求:

1. 实现一元二次方程类
2. 在主类中调用这个一元二次方程类。(编程用命令行或者可视化方法获取系数 a,b,c)

2.2 答案

2.2.1 项目结构树

```
D:.
└─src
    └─edu
        └─wanpengxu
            └─homework3
                └─second
                        AWrongException.java
                        Equation.java
                        Test.java
```

2.2.2 代码

2.2.2.1 核心代码

Equation.java

```
import java.io.Serializable;
import java.lang.Math;

public class Equation implements Serializable {
    private final double a, b, c;
    private double x1, x2;
    private double r, v;  // r:实部 v:虚部
    int type;

    public Equation(double a, double b, double c) throws AWrongException {
        if (a == 0) throw new AWrongException("二次项系数不能为0! ");
        this.a = a;
        this.b = b;
        this.c = c;
        System.out.println("成功创建一元二次方程组: ");
        System.out.print((a == 1 ? "" : a) + "x^2");  // 系数为1时省略
        if (b != 0) System.out.print("+" + (b == 1 ? "" : b) + "x");
        if (c != 0) System.out.print("+" + c);
        System.out.println("=0");
    }

    private double calculateDelta() {
        return Math.pow(b, 2) - 4 * a * c;
    }

    public void solve() {
        double delta = calculateDelta();
        if (Math.abs(delta) < 1e-5) {
            type = 0;
            x1 = x2 = -b / (2 * a);
        } else if (delta > 0) {
            type = 1;
            x1 = (-b - Math.sqrt(delta)) / (2 * a);
            x2 = (-b + Math.sqrt(delta)) / (2 * a);
        } else {
            type = 2;
            r = -b / (2 * a);
            v = Math.sqrt(-delta) / (2 * a);
        }
    }

    public void display() {
        switch (type) {
```

```
            case 0 -> System.out.println("该一元二次方程有两个相等的实数根：\nx1=x2=" + x1);
            case 1 -> System.out.println("该一元二次方程有两个不同的实数根：\nx1=" + x1 + '\n'
+ "x2=" + x2);
            case 2 -> System.out.printf("该一元二次方程有两个不同的复数根：
\nx1=%1$s+%2$si\nx2=%1$s-%2$si\n", r, v);
        }
    }

    public String getSolution() {
        return switch (type) {
            case 0 -> String.format("该一元二次方程有两个相等的实数根：\nx1=x2=%s", x1);
            case 1 -> String.format("该一元二次方程有两个不同的实数根：\nx1=%s\nx2=%s", x1,
x2);
            case 2 -> String.format("该一元二次方程有两个不同的复数根：
\nx1=%1$s+%2$si\nx2=%1$s-%2$si\n", r, v);
            default -> throw new IllegalStateException("Unexpected value: " + type);
        };
    }
}
```

AWrongException.java

```
package edu.wanpengxu.homework3.second;

public class AWrongException extends Exception{  // 系数a错误异常
    AWrongException(String s){
        super(s);
    }
}
```

2.2.2.2 控制台程序

Test.java

```
package edu.wanpengxu.homework3.second;

import java.io.*;
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        try (Scanner sc = new Scanner(System.in)) {
            System.out.println("请输入一元二次方程的系数：");
            double a = sc.nextDouble();
            double b = sc.nextDouble();
            double c = sc.nextDouble();
            try {
                Equation equation = new Equation(a, b, c);
```

```
                equation.solve();
                equation.display();
                try (FileOutputStream fileOutputStream = new FileOutputStream("file1.data");
                     ObjectOutputStream objectOutputStream = new
ObjectOutputStream(fileOutputStream)) {
                    objectOutputStream.writeObject(equation);
                    System.out.println("已将一元二次方程对象序列化为byte数组保存至文件
\"file1.data\"中");
                } catch (IOException e) {
                    System.out.println(e.getMessage());
                }
            } catch (AWrongException e) {
                System.out.println(e.getMessage());
                e.printStackTrace();
            }
        }
        try (FileInputStream fileInputStream = new FileInputStream("file1.data");
             ObjectInputStream objectInputStream = new ObjectInputStream(fileInputStream)) {
            Equation equation = (Equation) objectInputStream.readObject();
            System.out.println("已将文件\"file1.data\"中的byte数组反序列化为一元二次方程对象");
            equation.display();
        } catch (IOException | ClassNotFoundException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

2.2.2.3 GUI程序

```
import com.formdev.flatlaf.FlatLightLaf;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;

public class UnivariateQuadraticEquation {
    private JPanel mainPanel;
    private JTextField textFieldA;
    private JTextField textFieldB;
    private JTextField textFieldC;
    private JButton 求解Button;
    private JButton 序列化至文件Button;
    private JButton 清空Button;
    private JButton 退出Button;

    public static void main(String[] args) {
        try {
```

```
            UIManager.setLookAndFeel(new FlatLightLaf());
        } catch (Exception ex) {
            System.err.println("Failed to initialize LaF");
        }

        JFrame frame = new JFrame("一元二次方程综合工具");

        final Toolkit toolkit = Toolkit.getDefaultToolkit();
        toolkit.addAWTEventListener(e -> {
            if (e.getID() == KeyEvent.KEY_PRESSED) {
                KeyEvent evt = (KeyEvent) e;
                if (evt.getKeyCode() == KeyEvent.VK_ESCAPE) {
                    frame.dispose();  // 销毁当前JFrame
                    System.exit(0);  // 终止当前程序的JVM
                }
            }
        }, AWTEvent.KEY_EVENT_MASK);

        Image icon = toolkit.getImage("img/CUMTlogo5.png");
        frame.setIconImage(icon);
        frame.setContentPane(new UnivariateQuadraticEquation().mainPanel);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
//        frame.setSize(400, 130);
        frame.pack();
        frame.setLocationRelativeTo(null);  // 这句话要放在形状设置后，否则是将size(0,0)的窗口
居中，造成窗口左上角在中间的情况
        frame.setResizable(false);
//        frame.getContentPane().setBackground(Color.white);
        frame.setVisible(true);
    }

    public UnivariateQuadraticEquation() {
        // lambda表达式
        求解Button.addActionListener(e -> {
            try {
                double a = Double.parseDouble(textFieldA.getText());
                double b = Double.parseDouble(textFieldB.getText());
                double c = Double.parseDouble(textFieldC.getText());
                Equation equation = new Equation(a, b, c);
                equation.solve();
                equation.display();
                JOptionPane.showMessageDialog(null, equation.getSolution(), "成功",
JOptionPane.INFORMATION_MESSAGE);
            } catch (NumberFormatException e1) {
                JOptionPane.showMessageDialog(null, "请输入合法的系数！", "异常",
JOptionPane.WARNING_MESSAGE);
            } catch (AWrongException e2) {
```

```
                JOptionPane.showMessageDialog(null, e2.getMessage(), "异常",
JOptionPane.WARNING_MESSAGE);
//                e2.printStackTrace();
            }
        });
        // 匿名内部类
        序列化至文件Button.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                try (FileOutputStream fileOutputStream = new FileOutputStream("file1.data");
ObjectOutputStream objectOutputStream = new ObjectOutputStream(fileOutputStream)) {
                    double a = Double.parseDouble(textFieldA.getText());
                    double b = Double.parseDouble(textFieldB.getText());
                    double c = Double.parseDouble(textFieldC.getText());
                    Equation equation = new Equation(a, b, c);
                    objectOutputStream.writeObject(equation);
                    JOptionPane.showMessageDialog(null, "已将一元二次方程对象序列化为byte数组保
存至文件\"file1.data\"中", "消息提示", JOptionPane.INFORMATION_MESSAGE);
                } catch (NumberFormatException e1) {
                    JOptionPane.showMessageDialog(null, "请输入合法的系数！", "异常",
JOptionPane.WARNING_MESSAGE);
                } catch (IOException | AWrongException e2) {
                    // 这里用word打开可以测试占用
                    JOptionPane.showMessageDialog(null, e2.getMessage(), "异常",
JOptionPane.WARNING_MESSAGE);
                    e2.printStackTrace();
                }
            }
        });
        清空Button.addActionListener(e -> {
            textFieldA.setText("");
            textFieldB.setText("");
            textFieldC.setText("");
        });
        退出Button.addActionListener(e -> System.exit(0));
    }
}
```

2.2.3 运行截图

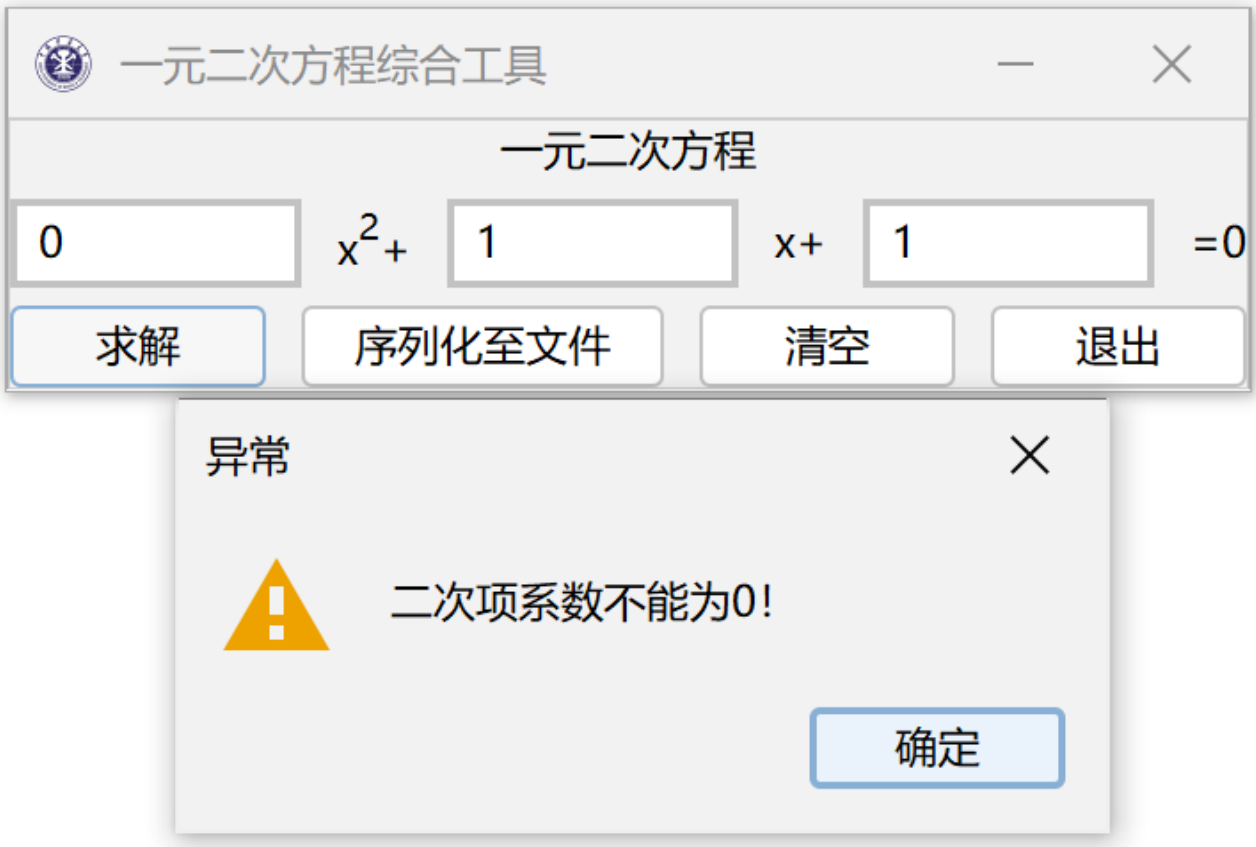
2.2.3.1 控制台程序

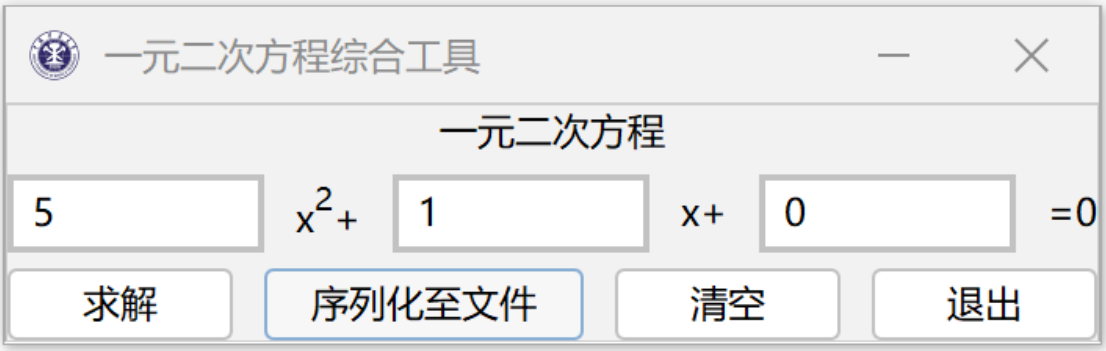
```
请输入一元二次方程的系数：
0 1 1
二次项系数不能为0!
edu.wanpengxu.homework3.second.AWrongException Create breakpoint : 二次项系数不能为0!
    at edu.wanpengxu.homework3.second.Equation.<init>(Equation.java:13)
    at edu.wanpengxu.homework3.second.Test.main(Test.java:14)
已将文件"file1.data"中的byte数组反序列化为一元二次方程对象
该一元二次方程有两个不同的实数根：
x1=-0.2
x2=0.0

进程已结束,退出代码0
```

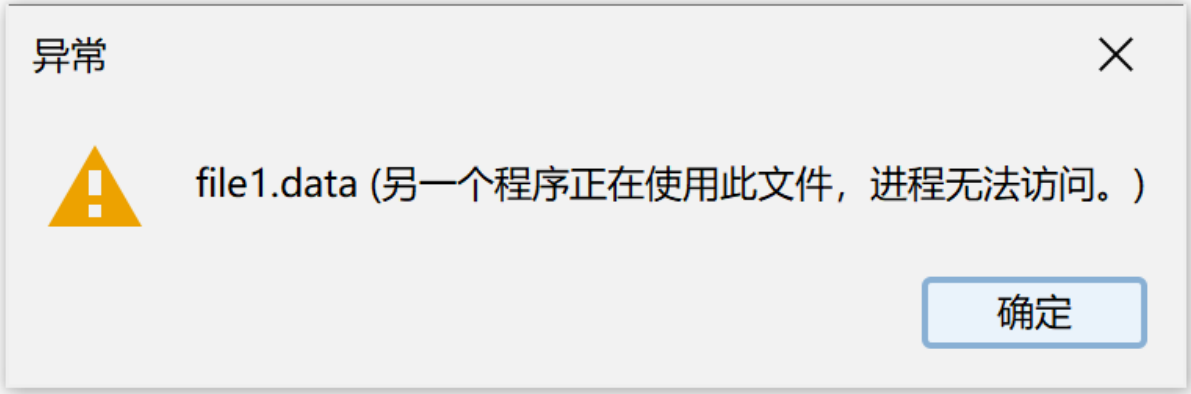
```
请输入一元二次方程的系数：
5 1 0
成功创建一元二次方程组：
5.0x^2+x=0
该一元二次方程有两个不同的实数根：
x1=-0.2
x2=0.0
已将一元二次方程对象序列化为byte数组保存至文件"file1.data"中
已将文件"file1.data"中的byte数组反序列化为一元二次方程对象
该一元二次方程有两个不同的实数根：
x1=-0.2
x2=0.0

进程已结束,退出代码0
```

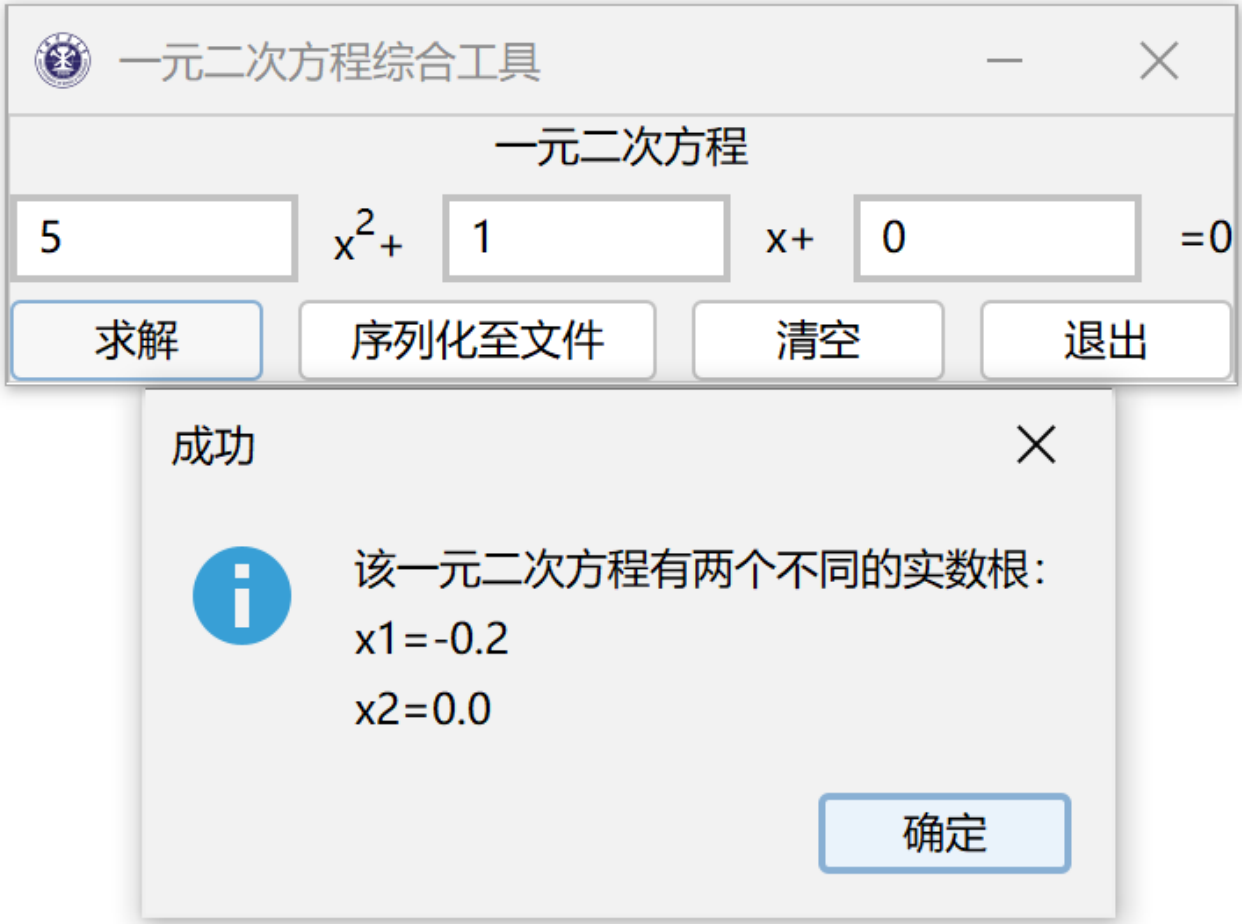
```
请输入一元二次方程的系数：
5 2 1
成功创建一元二次方程组：
5.0x^2+2.0x+1.0=0
该一元二次方程有两个不同的复数根：
x1=-0.2+0.4i
x2=-0.2-0.4i
已将一元二次方程对象序列化为byte数组保存至文件"file1.data"中
已将文件"file1.data"中的byte数组反序列化为一元二次方程对象
该一元二次方程有两个不同的复数根：
x1=-0.2+0.4i
x2=-0.2-0.4i

进程已结束,退出代码0
```

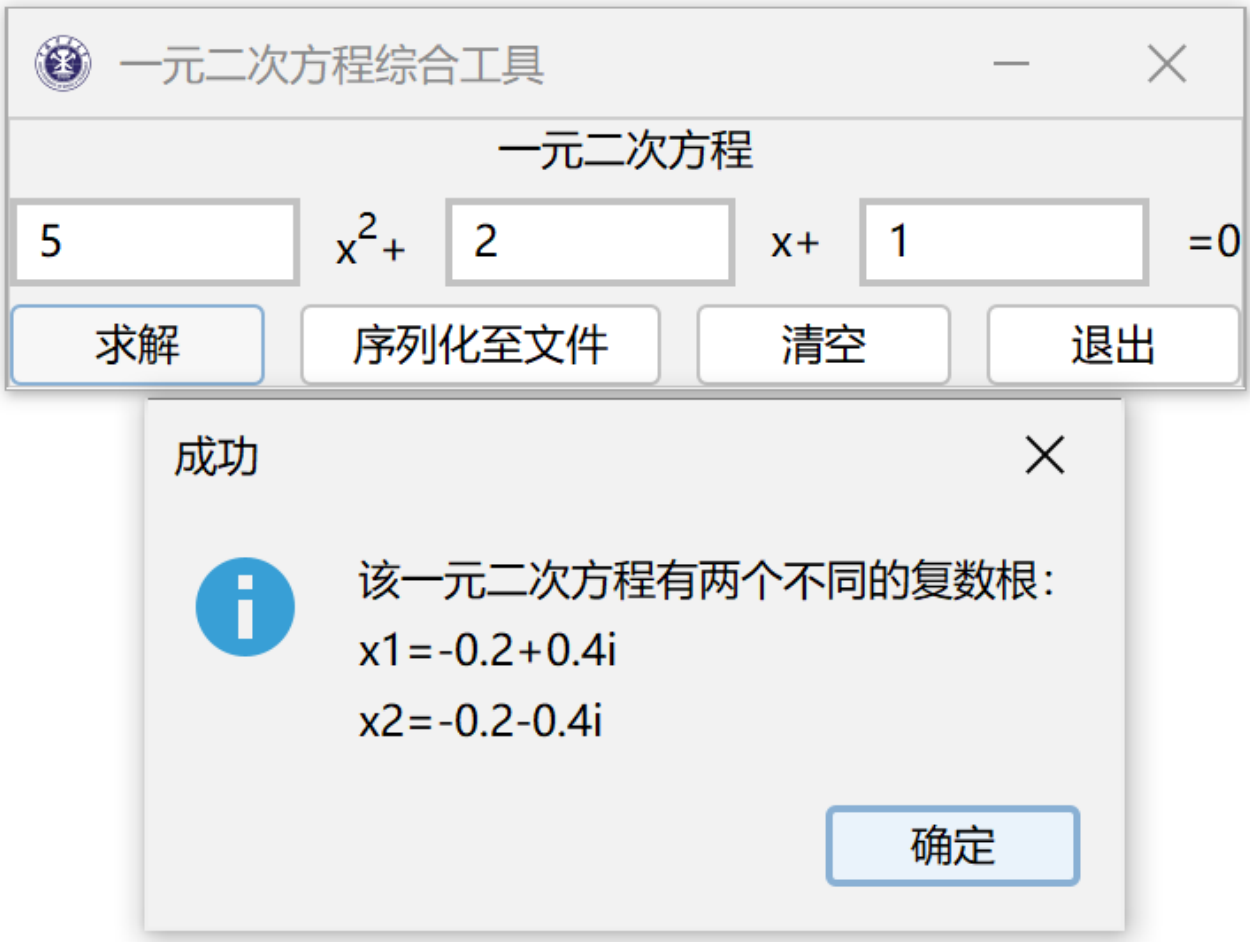
2.2.3.2 GUI程序




一元二次方程综合工具
一元二次方程
5
x2+
1
x+
0
=0
求解
序列化至文件
清空
退出


异常
file1.data (另一个程序正在使用此文件，进程无法访问。)
确定


一元二次方程综合工具
一元二次方程
5
x2+
1
x+
0
=0
求解
序列化至文件
清空
退出
成功
该一元二次方程有两个不同的实数根:
x1=-0.2
x2=0.0
确定



2.3 分析

模拟题，用一些基本的数学知识可以写出核心代码，这里我认为解的形式都应该是`double`，如果将复数解作为一个整体，那么要么用`String`表示，要么自己写一个复数类，前者不那么专业，后者使程序复杂，所以我认为将实部（real part）和虚部（imaginary part）拆开都用基本数据类型`double`表示更优。

再用一些课上的知识可以写出控制台程序

GUI程序需要自己学习，不过在周三的课上已经讲了很大一部分，包括监听器的三种回调方式，所以我在课内基本目标外实现的主要有：

1. 按ESC键退出，知识是任意componet监听，详见GUI程序中的`toolkit.addAWTEventListener(...)`。
2. 程序自定义图标，调`JFrame.setIconImage(Image)`即可，但将其打包到JAR中有难度。
3. 界面美化，swing支持第三方主题包，我使用的是`Intelli IDEA`同款主题`FlatLaf`，使用方法是在项目中导入JAR，之后使用初始化语句`UIManager.setLookAndFeel(new FlatLightLaf());`即可。另外，可将LookAndFeel译为“观感”。