



华南理工大学
South China University of Technology

JAVA 实验报告

学 院 计算机科学与工程学院

专 业 信息安全

学生姓名 颜徐柳

学生学号 201630610779

指导教师 布社辉

课程编号 145210

课程学分 2.5

起始日期 2017/11/1

| | |
|--|----|
| 实验 1：二分法求函数的根 | 4 |
| - 实验题目 | 4 |
| - 问题分析 | 4 |
| - 实验的具体步骤 | 4 |
| - 实现程序的完整代码 | 5 |
| - 实验结果以及总结 | 6 |
| 实验 2：求 10,000,000 内最大的素数 | 7 |
| - 实验题目 | 7 |
| - 问题分析 | 7 |
| - 实验的具体步骤 | 7 |
| - 实现程序的完整代码 | 8 |
| - 实验结果以及总结 | 8 |
| 实验 3：K-MEANS 算法实现对数据的聚类分析 | 10 |
| - 实验题目 | 10 |
| - 问题分析 | 10 |
| - 实验的具体步骤 | 10 |
| - 实现程序的完整代码 | 11 |
| - 实验结果以及总结 | 11 |
| 实验结果： | 13 |
| 总结： | 13 |
| 实验 4：线性回归(LINEAR REGRESSION)找到最佳拟合直线 | 14 |
| - 实验题目 | 14 |
| - 问题分析 | 15 |
| - 实验的具体步骤 | 15 |
| - 实现程序的完整代码 | 15 |
| - 实验结果以及总结 | 16 |
| 实验 5：统计学生成绩 | 18 |
| - 实验题目 | 18 |
| - 问题分析 | 18 |
| - 实验的具体步骤 | 18 |
| - 实现程序的完整代码 | 18 |
| - 实验结果以及总结 | 19 |
| 实验 6：图形用户界面 | 21 |
| - 实验题目 | 21 |
| - 问题分析 | 21 |
| - 实验的具体步骤 | 22 |
| - 实现程序的完整代码 | 22 |
| - 实验结果以及总结 | 26 |
| 实验 7：JDBC 方式操作数据库 | 30 |
| - 实验题目 | 30 |
| - 问题分析 | 31 |
| - 实验的具体步骤 | 31 |
| - 实现程序的完整代码(很长) | 31 |
| - 实验结果以及总结 | 38 |

| | |
|---------------------|----|
| 实验 8：网络编程基础 | 42 |
| - 实验题目 | 42 |
| - 问题分析 | 42 |
| - 实验的具体步骤 | 42 |
| - 实现程序的完整代码 | 42 |
| - 实验结果以及总结 | 47 |
| 实验 10：重复文件的查询 | 48 |
| - 实验题目 | 48 |
| - 问题分析 | 48 |
| - 实验的具体步骤 | 48 |
| - 实现程序的完整代码 | 48 |
| - 实验结果以及总结 | 52 |

实验 1：二分法求函数的根

- 实验题目

##实验 1：二分法求函数的根

###函数

\$\$

$f(x) = x^3 - 10x + 23$

\$\$

###初始值

- $x_{\text{low}} = -10.0$

- $x_{\text{high}} = 5.0$

- $\delta = 0.001$ ，即 $|x_{\text{high}} - x_{\text{low}}| \leq \delta$

###二分法

1. $x_{\text{high}} \cdot x_{\text{low}} < 0$

2. $x_c = \frac{x_{\text{high}} + x_{\text{low}}}{2}$

3. 当 $f(x_c) = 0$ 时，即得到根值，将 x_c 作为结果输出打印并退出程序

4. 否则，当 $f(x_c) \neq 0$ 时，执行下述步骤 5 或 6

5. 当 $x_{\text{high}} \cdot x_c < 0$ 时， $x_{\text{low}} = x_c$

6. 当 $x_{\text{low}} \cdot x_c < 0$ 时， $x_{\text{high}} = x_c$

7. 当 $|x_{\text{high}} - x_{\text{low}}| \leq \delta$ 时，将 x_c 作为结果输出打印并退出程序，否则执行步骤 2

。

- 问题分析

需要解决的技术问题：求函数的根

采用的技术方案：二分法

- 实验的具体步骤

用文本编辑器完成代码：

1. 建立 Division 类；
2. 建立静态方法 f 计算函数值；

3. 静态 main 方法完成二分求根。

- 实现程序的完整代码

```
import java.io.*;

public class Division {

    private static double f(double x) {
        return x*x*x - x*10 + 23;
    }

    public static void main(String[] args) {
        try {
            PrintStream ps = new PrintStream(new
FileOutputStream("out/01-division.txt"));
            System.setOut(ps);
        } catch (IOException e) {
            System.out.println("Failed to open file!");
            System.exit(-1);
        }
        double delta = 0.001;
        double x1 = -10.0;
        double x2 = 5.0;
        while (true) {
            double x = (x1 + x2) / 2.0;
            double y1 = f(x1);
            double y2 = f(x2);
            double y = f(x);
            if (x2 - x1 < delta || y == 0) {
                System.out.println(x);
                System.exit(0);
            }
            if (y2 * y < 0)
                x1 = x;
            else
                x2 = x;
        }
    }
}
```

- 实验结果以及总结

实验结果：

out/01-division.txt

-3.973541259765625

总结：二分法可快速寻找函数的根

实验 2：求 10,000,000 内最大的素数

- 实验题目

##实验 2：求 10,000,000 内最大的素数

###参考程序：求解 100 以内的全部素数

~~~java

```
public class PrimeApp {
    public static void main(String[] args) {
        int m, n; // 变量 n 为要判断的数字
        System.out.println("100 以内的素数有：");
        A: for (n = 2; n <= 100; n++) {
            for (m = 2; m < n / 2; m++) { // 【此行应为 for (m = 2; m <= n / 2; m++) { 】
                if (n % m == 0)
                    continue A;
                // 如果能被整除则变量 n 肯定不是素数，跳出内层循环
            }

            System.out.print(n + " " + "\t");//输出素数
        }
    }
}
```

~~~

- 问题分析

需要解决的技术问题：数据过大，无法用平凡方法求出
以及采用的技术方案：埃拉托斯筛法求素数

- 实验的具体步骤

用文本编辑器完成以下代码

- 实现程序的完整代码

```
import java.io.*;

public class Prime {

    public static void calc(boolean[] list, int n, int a, int b) {
        int i=b;
        while(i<a) {
            while(!list[++i]);
            int j = a / i;
            j = j<2 ? 2 : j;
            for (;i*j<=n;++j) {
                list[i*j] = false;
            }
        }
        if (a*a < n)
            calc(list, n, a*a, a);
    }

    public static void main(String args[]) {
        try {
            PrintStream ps = new PrintStream(new
FileOutputStream("out/02-prime.txt"));
            System.setOut(ps);
        } catch (IOException e) {
            System.out.println("Failed to open file!");
            System.exit(-1);
        }
        int n = 10000000;
        boolean[] list = new boolean[n+1];
        int[] tmp = {0,0,1,1,0,1,0,1,0,0,0,1,0,1,0,0,0,1,0,1,0,0,0,1};
        for (int i=1;i<=n;++i)
            list[i] = (i<24 && tmp[i]==0) ? false : true;
        calc(list, n, 23, 1);
        while (!list[--n]);
        System.out.println(n);
    }
}
```

- 实验结果以及总结

实验结果：

out/02-prime.txt

9999991

总结:

用埃拉托斯筛法求素数

实验 3：K-Means 算法实现对数据的聚类分析

- 实验题目

。##实验 3：K-Means 算法实现对数据的聚类分析

K-Means 算法的基本思想是初始随机给定 K 个簇中心，按照最邻近原则把待分类样本点分到各个簇。

然后按平均法重新计算各个簇的质心，从而确定新的簇心。

一直迭代，直到簇心的移动距离小于某个给定的值。

###工作原理是：

~~~

选择 K 个点作为初始质心(随机选择)

Repeat

    计算数据集中每个点到各个簇的质心的距离，将数据点分配到距离最近的簇

    重新计算每个簇中所有点的均值并将该均值作为质心

Until 簇不发生变化或达到最大迭代次数

~~~

- 其中 K 是用户指定的参数，即所期望的簇的个数。
- 常用的距离度量方法为：欧几里得距离
- 簇的质心都是其均值，即向量各维取平均即可

###Data Sets:

- KMeans_Set.txt
- KMeans_Set2.txt

- 问题分析

用题中所给算法完成算法设计

- 实验的具体步骤

用文本编辑器完成以下代码

- 实现程序的完整代码

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

class Point {
    public double x;
    public double y;
    public Point(double x, double y) {
        this.x = x;
        this.y = y;
    }
}

public class KMeans {

    private int n;
    private Point[] data;

    private static Point[] create_keys(int n) {
        Point[] keys = new Point[n];
        Random rand = new Random(System.currentTimeMillis());
        for (int i=0;i<keys.length;++i)
            keys[i] = new Point(rand.nextDouble(), rand.nextDouble());
        return keys;
    }

    private static double d(Point a, Point b) {
        return (a.x-b.x)*(a.x-b.x) + (a.y-b.y)*(a.y-b.y);
    }

    public KMeans(String filename) throws FileNotFoundException {
        data = new Point[100];
        n = 0;
        Scanner sc = new Scanner(new File(filename));
        while (sc.hasNextDouble()) {
            double x = sc.nextDouble();
            double y = sc.nextDouble();
            data[n++] = new Point(x, y);
        }
        System.out.println(filename);
        System.out.println("-----");
    }
}
```

```

}

public KMeans calc(int loop, Point keys[]) {
    Point[] k = new Point[keys.length];
    int[] nn = new int[keys.length];
    for (int i=0;i<k.length;++i) {
        nn[i] = 0;
        k[i] = new Point(0, 0);
    }
    for (int i=0;i<n;++i) {
        double mind = -1;
        int minp = -1;
        for (int j=0;j<keys.length;++j) {
            double tmp = d(data[i], keys[j]);
            if (minp == -1 || mind > tmp) {
                mind = tmp;
                minp = j;
            }
        }
        k[minp].x += data[i].x;
        k[minp].y += data[i].y;
        nn[minp] ++;
    }
    boolean is_change = false;
    for (int i=0;i<keys.length;++i) {
        if (nn[i] == 0)
            continue;
        double x = k[i].x / nn[i];
        double y = k[i].y / nn[i];
        if (keys[i].x != x || keys[i].y != y)
            is_change = true;
        keys[i].x = x;
        keys[i].y = y;
    }
    if (is_change && loop > 1)
        calc(loop - 1, keys);
    return this;
}

public void print(Point keys[]) {
    for (int i=0;i<keys.length;++i)
        System.out.format("%f, %f\r\n", keys[i].x, keys[i].y);
    System.out.println("");
}

```

```

    public static void main(String args[]) {
        Point[] key1 = create_keys(4);
        Point[] key2 = create_keys(3);
        try {
            PrintStream ps = new PrintStream(new
FileOutputStream("out/03-kMeans.txt"));
            System.setOut(ps);
            new KMeans("data/KMeans_Set.txt").calc(10,
key1).print(key1);
            new KMeans("data/KMeans_Set2.txt").calc(10,
key2).print(key2);
        } catch (IOException e) {
            System.out.println("Failed to open file!");
            System.exit(-1);
        }
    }
}

```

- 实验结果以及总结

实验结果:

out/03-kMeans.txt

data/KMeans_Set.txt

2.802931, -2.731515

-3.382370, -2.947336

2.626530, 3.108680

-2.461543, 2.787376

data/KMeans_Set2.txt

2.933864, 3.127828

-0.459656, -2.778216

-2.947376, 3.326378

总结:

数据的归类与统计是很重要的算法

实验 4：线性回归(Linear Regression)找到最佳拟合直线

- 实验题目

##实验 4：线性回归(Linear Regression)找到最佳拟合直线

在统计学中，线性回归(Linear Regression)是利用称为线性回归方程的最小平方函数对一个或多个自变量和因变量之间关系进行建模的一种回归分析。

设直线方程为

\$\$

$$f(x) = A x + B$$

\$\$

设数据集为 $\{(x_k, y_k)\}_{k=1}^N$

误差均值为：

$$E_2(f)^2 = \frac{1}{N} \sum_{k=1}^N (Ax_k + B - y_k)^2$$

若

\$\$

$$\frac{\partial E(A, B)}{\partial A} = 0$$

\$\$

\$\$

$$\frac{\partial E(A, B)}{\partial B} = 0$$

\$\$

则

\$\$

$$0 = \sum_{k=1}^N (Ax_k^2 + Bx_k - x_k y_k) = A \sum_{k=1}^N x_k^2 + B \sum_{k=1}^N x_k - \sum_{k=1}^N x_k y_k$$

\$\$

\$\$

$$0 = \sum_{k=1}^N (Ax_k + B - y_k) = A \sum_{k=1}^N x_k + NB - \sum_{k=1}^N y_k$$

\$\$

即：

\$\$

$$\left(\sum_{k=1}^N x_k^2 \right) A + \left(\sum_{k=1}^N x_k \right) B = \sum_{k=1}^N x_k y_k$$

\$\$

\$\$

$$\left(\sum_{k=1}^N x_k \right) A + N B = \sum_{k=1}^N y_k$$

最后根据上述两式可以求出\$A\$和\$B\$

###Data Sets:

- LR_ex0.txt
- LR_ex1.txt

- 问题分析

需要解决的技术问题：求线性回归 A、B

采用的技术方案：线性回归方程

- 实验的具体步骤

用文本编辑器完成以下代码

- 实现程序的完整代码

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

class Point {
    public double x;
    public double y;
    public Point(double x, double y) {
        this.x = x;
        this.y = y;
    }
}

public class Linear {

    private static void linear(String filename) throws
FileNotFoundException {
        Vector<Point> v = new Vector<Point>();
        Scanner sc = new Scanner(new File(filename));
        System.out.println(filename);
```

```

System.out.println("-----");
Point total = new Point(0, 0);
while (sc.hasNextDouble()) {
    sc.nextDouble();
    double x = sc.nextDouble();
    double y = sc.nextDouble();
    v.add(new Point(x, y));
    total.x += x;
    total.y += y;
}
Point avg = new Point(total.x/v.size(), total.y/v.size());
double sub = 0.0;
double sup = 0.0;
for (int i=0;i<v.size();++i) {
    Point p = v.get(i);
    sub += (p.x - avg.x) * (p.y - avg.y);
    sup += (p.x - avg.x) * (p.x - avg.x);
}
double a = sub / sup;
double b = avg.y - a * avg.x;
System.out.format("y = %fx + %f\r\n\r\n", a, b);
}

public static void main(String[] args) {
    try {
        PrintStream ps = new PrintStream(new
FileOutputStream("out/04-linear.txt"));
        System.setOut(ps);
        linear("data/LR_ex0.txt");
        linear("data/LR_ex1.txt");
    } catch (IOException e) {
        System.out.println("Failed to open file!");
        System.exit(-1);
    }
}
}

```

- 实验结果以及总结

结果:

out/04-linear.txt

data/LR_ex0.txt

$$y = 1.695323x + 3.007743$$

data/LR_ex1.txt

$$y = 1.668743x + 3.007722$$

总结:

数学很重要

实验 5：统计学生成绩

- 实验题目

##实验 5：统计学生成绩

###要求：

- 从指定的文本文件读取成绩
- 获取最高成绩和最低成绩
- 计算所有学生的平均成绩
- 分别统计\$60\$~\$69\$, \$70\$~\$79\$, \$80\$~\$89\$, \$90\$~\$100\$分数段的学生人数
- 所有结果要输出到一个文本文件

###Data Sets:

- score.csv

- 问题分析

需要解决的技术问题：文件输入输出

采用的技术方案：文件输入输出流

*测试前已将输入文件 `data/score.csv` 转换成标准逗号分隔符 `csv` 文件

- 实验的具体步骤

用文本编辑器完成以下代码

- 实现程序的完整代码

```
import java.io.*;

public class Score {
    public static void main(String[] args) {
        int total = 0;
        int n = 0;
        int max = 0, min = 100;
        int[] part = new int[11];
```

```

    try {
        PrintStream ps = new PrintStream(new
FileOutputStream("out/05-score.txt"));
        System.setOut(ps);
        BufferedReader in = new BufferedReader(new
FileReader("data/score.csv"));
        String line = null;
        while ((line = in.readLine()) != null) {
            String[] items = line.split(",");
            int score = Integer.parseInt(items[items.length-1]);
            total += score;
            n ++;
            part[score/10] ++;
            if (score > max)
                max = score;
            if (score < min)
                min = score;
        }
        System.out.format("The highest grade: %d\r\n", max);
        System.out.format("The lowest grade: %d\r\n", min);
        System.out.format("Average grade: %d\r\n", total/n);
        System.out.format("Number of grade 60-69: %d\r\n",
part[6]);
        System.out.format("Number of grade 70-79: %d\r\n",
part[7]);
        System.out.format("Number of grade 80-89: %d\r\n",
part[8]);
        System.out.format("Number of grade 90-100: %d\r\n",
part[9]+part[10]);
    } catch (IOException e) {
        System.out.println("Failed to open file!");
        System.exit(-1);
    }
}
}

```

- 实验结果以及总结

实验结果:

out/05-score.txt

The highest grade: 97

The lowest grade: 22

Average grade: 69

Number of grade 60-69: 14

Number of grade 70-79: 13

Number of grade 80-89: 11

Number of grade 90-100: 3

总结:

文件输入输出很重要

实验 6：图形用户界面

- 实验题目

。##实验 6：图形用户界面

###模拟计算器

复习教课书上的模拟计算器的程序例子，并实现该程序

###五子棋的游戏

实验要求：

- 棋盘为 10 行 10 列，棋盘底色可自己设定，棋盘格子间有一定间隙。
- 本游戏为单机版，故而为了模拟双方下棋的效果，可一人下棋时显示“O”，另一人下棋时显示“X”，以示区分并相互交替。并且，一个格子如果已经下过棋子，则不可再改变它上的棋子，即不能再更改其上显示的字符，或者说按钮变得不可用。如下图所示。
- 游戏规则：
当同一行或同一列或同一条对角线上有连续的五个相同棋子时判赢。若最终棋盘满了仍未有五个相同棋子连成一线，则判输。若未有这两种情况，则继续下棋。（注：此功能需要一定的算法设计和数据结构知识，如果时间不足，可不做）
- 无论最终是赢或输，都自动弹出一个消息框，并给出相应的提示，如下图所示。（注：可将本要求简化为，针对棋盘中的某两个特定格子，一个格子单击后弹出成功对话框，一个格子弹出失败对话框）

- 问题分析

需要解决的技术问题：图形化界面、网络传输（网络版）

采用的技术方案：awt、swing、双缓冲防闪烁（网络版）、NIO（网络版）

*计算器涉及太多逻辑算法，这里仅实现了五子棋的任务

*五子棋做了两个版本，一个单机版，一个网络版（重点放在网络版的 NIO 上，其他部分比较简陋）

*由于网络联机版包含文件过多（约有 20 个 Java 类），文档难以全部给出，这里仅给出结果截图，如需查看源代码请移步 github 项目地址：<https://github.com/x-insane/java-gobang>

- 实验的具体步骤

用文本编辑器完成以下代码（单机版）

- 实现程序的完整代码

```
import java.awt.Graphics;
import java.awt.Rectangle;
import java.awt.Color;
import java.awt.event.MouseEvent;
import java.awt.event.MouseAdapter;
import javax.swing.JFrame;
import javax.swing.JOptionPane;

enum Player {
    None, Black, Blue;

    public static Player op(Player p) {
        if (p == Player.Black)
            return Player.Blue;
        if (p == Player.Blue)
            return Player.Black;
        return Player.None;
    }
}

public class GoBang extends JFrame {

    private static final int nx = 18; // 列数
    private static final int ny = 12; // 行数
    private static final Rectangle r = new Rectangle(50, 75, 700,
500);
    private static final double dx = (double)r.width / nx;
    private static final double dy = (double)r.height / ny;
    private static final int gor = 15;
    private Player data[][];
    private Player last = Player.None;
    private int last_x = -1;
    private int last_y = -1;
    private int status = 0;

    public GoBang() {
        setSize(800, 625);
```

```

        setLocation(200, 50);
        setResizable(false);
        setTitle("五子棋");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        data = new Player[nx+1][ny+1];
        for (int i=0;i<=nx;++i)
            for (int j=0;j<=ny;++j)
                data[i][j] = Player.None;
        addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                if (status != 0) {
                    reset();
                    return;
                }
                java.awt.Point p = e.getPoint();
                if (p.x + dx/2 < r.x || p.y + dy/2 < r.y)
                    return;
                int x = (int)((p.x - r.x + dx/2) / dx);
                int y = (int)((p.y - r.y + dy/2) / dy);
                if (x > nx || y > ny)
                    return;
                if (data[x][y] != Player.None)
                    return;
                Player player = Player.op(last);
                if (player == Player.None)
                    player = Player.Black;
                Go(x, y, player);
            }
        });
    }

    public void reset() {
        status = 0;
        last = Player.None;
        last_x = last_y = -1;
        for (int i=0;i<=nx;i++)
            for (int j=0;j<=ny;j++)
                data[i][j] = Player.None;
        repaint();
    }

    public void Go(int x, int y, Player p) {
        data[x][y] = p;
    }

```

```

repaint();
last = p;
last_x = x;
last_y = y;
for (int i=0;i<=1;i++) {
    for (int j=-1;j<=1;j++) {
        if (i==0 && j==0)
            continue;
        if (i==0 && j==-1)
            continue;
        int n = 1;
        int px = x;
        int py = y;
        while(true) {
            px += i;
            py += j;
            if (px <= nx && py <= ny && data[px][py] == p)
                n++;
            else
                break;
        }
        px = x;
        py = y;
        while(true) {
            px -= i;
            py -= j;
            if (px >= 0 && py >= 0 && data[px][py] == p)
                n++;
            else
                break;
        }
        if (n >= 5) {
            status = -1;
            if (p == Player.Black)
                JOptionPane.showMessageDialog(this, "黑方胜！点
击屏幕重新开始", "游戏结束", JOptionPane.INFORMATION_MESSAGE);
            else
                JOptionPane.showMessageDialog(this, "蓝方胜！点
击屏幕重新开始", "游戏结束", JOptionPane.INFORMATION_MESSAGE);
            return;
        }
    }
}
for (int i=0;i<=nx;i++)

```



```

        for (int j=0;j<=ny;j++)
            if (data[i][j] == Player.None)
                return;
        status = -1;
        JOptionPane.showMessageDialog(this, "平局！点击屏幕重新开始", "
游戏结束", JOptionPane.INFORMATION_MESSAGE);
    }

    public void DrawGo(Graphics g) {
        for (int i=0;i<=nx;++i) {
            for (int j=0;j<=ny;++j) {
                if (data[i][j] == Player.None)
                    continue;
                else if (data[i][j] == Player.Black)
                    g.setColor(Color.black);
                else
                    g.setColor(Color.blue);
                g.fillOval((int)(r.x+i*dx-gor), (int)(r.y+j*dy-gor),
gor*2, gor*2);
            }
        }
    }

    public void DrawBoard(Graphics g) {
        g.setColor(Color.black);
        for (int i=0;i<=nx;++i)
            g.drawLine((int)(r.x+i*dx), (int)(r.y), (int)(r.x+i*dx),
(int)(r.y+r.height));
        for (int i=0;i<=ny;++i)
            g.drawLine((int)(r.x), (int)(r.y+i*dy),
(int)(r.x+r.width), (int)(r.y+i*dy));
    }

    @Override
    public void paint(Graphics g) {
        super.paint(g);
        DrawBoard(g);
        DrawGo(g);
    }

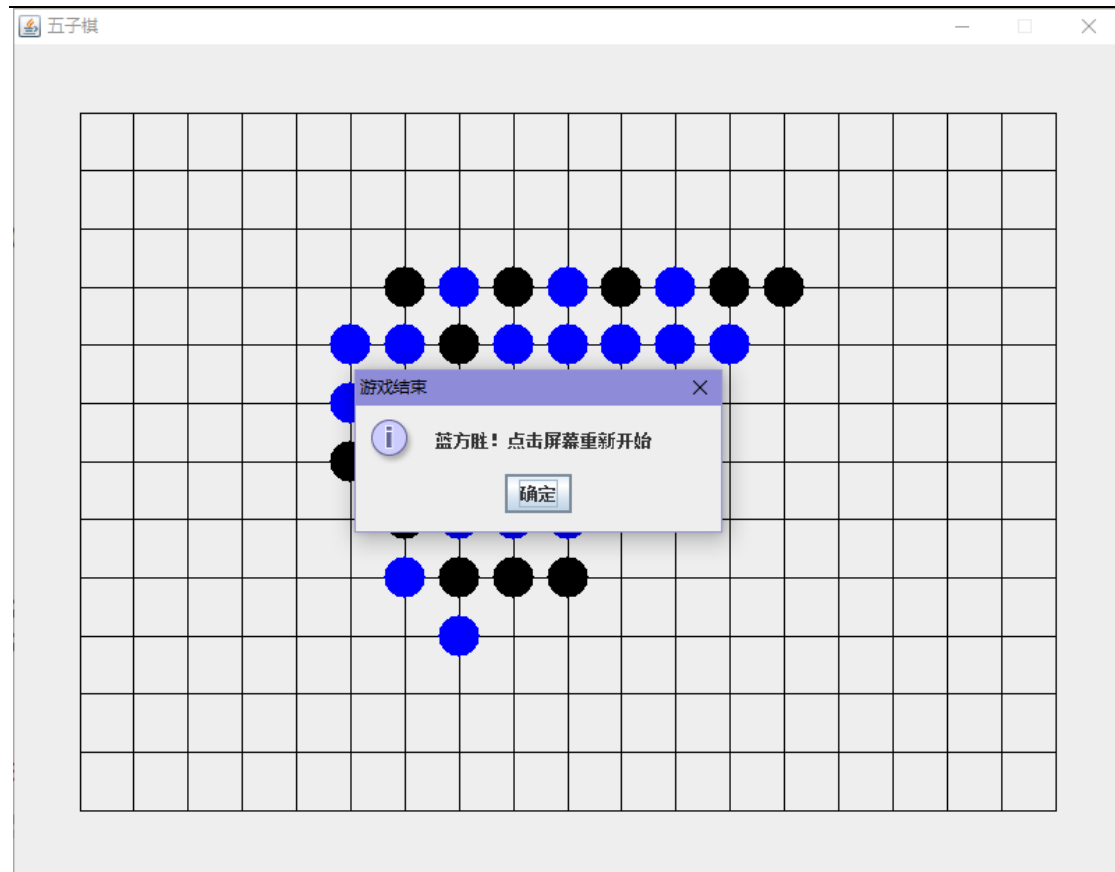
    public static void main(String[] args) {
        new GoBang().setVisible(true);
    }
}

```

- 实验结果以及总结

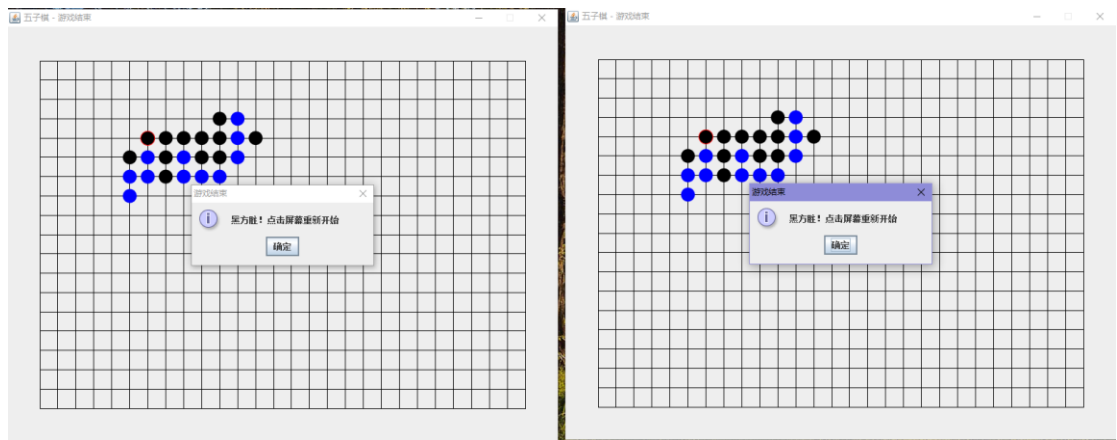
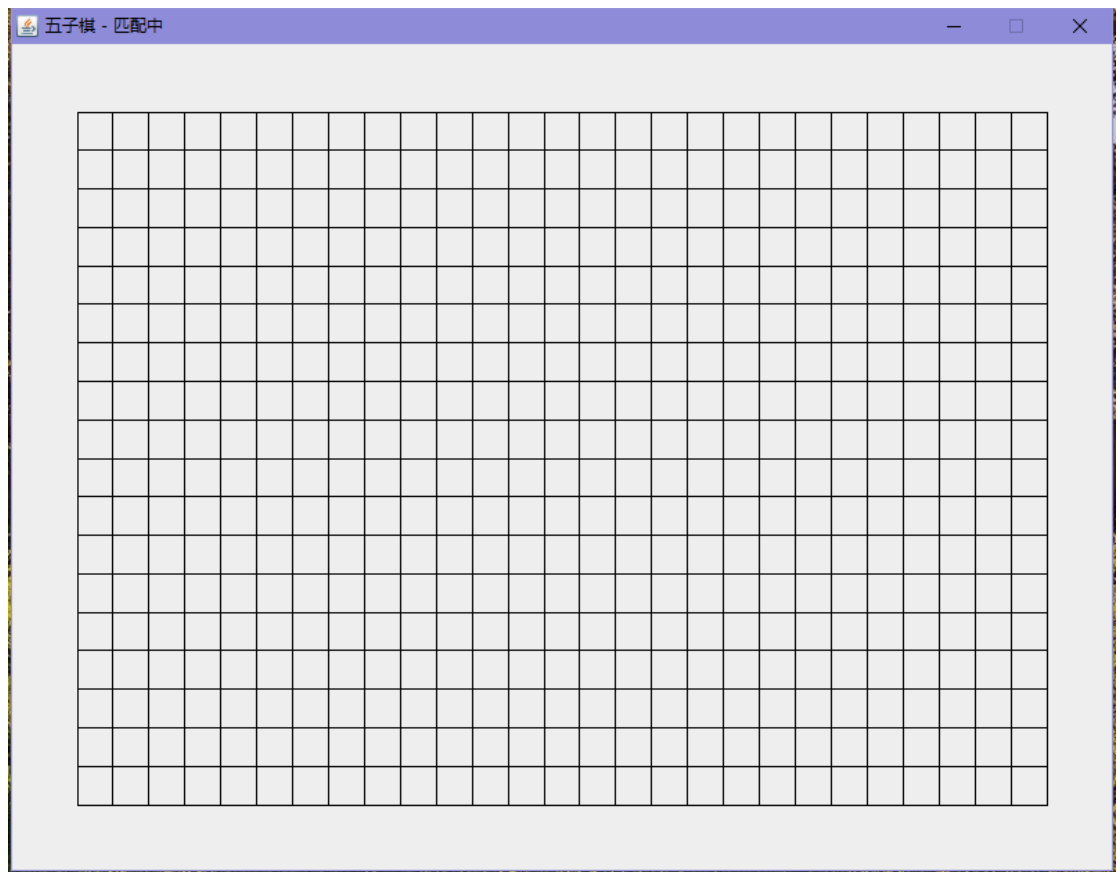
实验结果:

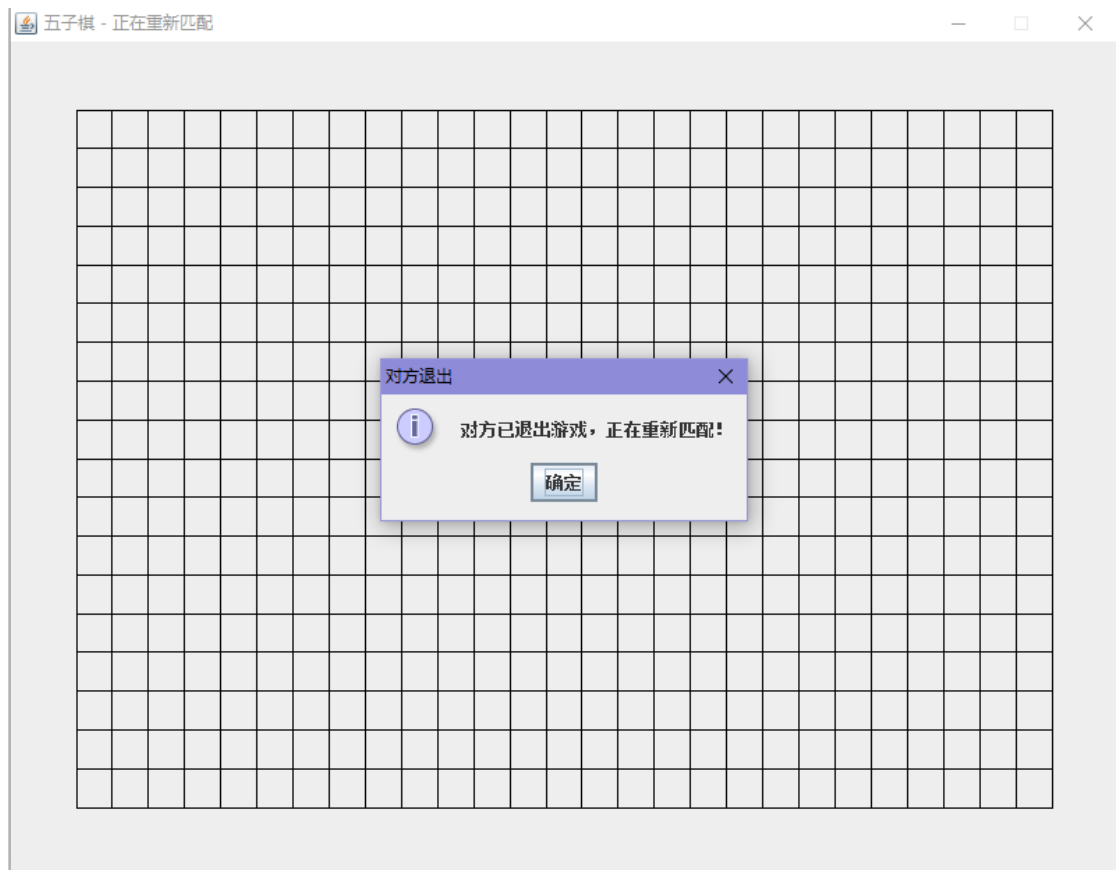
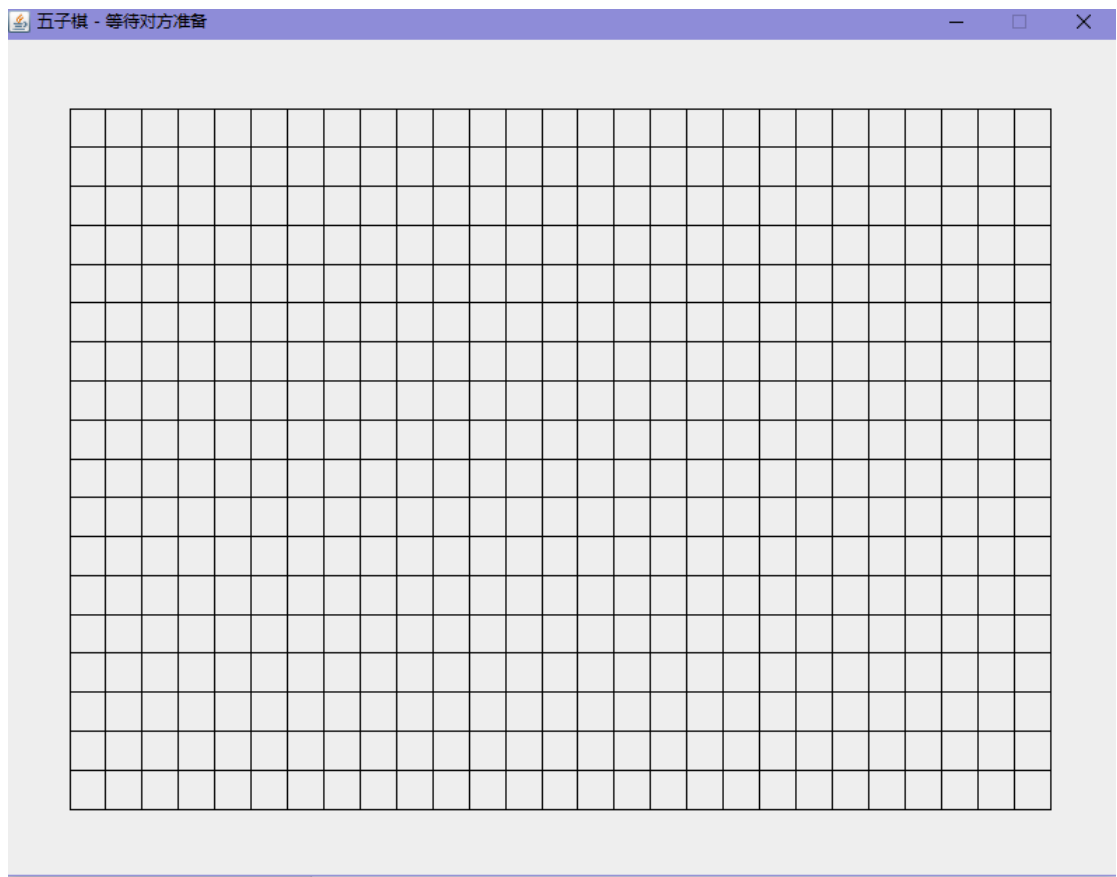
单机版



(早期版本, 未使用精细绘图, 未使用双缓冲)

网络联机版





总结:

之所以选择复杂得多的异步网络 IO，一是因为得知异步 IO 可以大大提高并发量，而是想挑战一下没接触过的部分。总体上讲，还有很多地方值得进一步研究，由于时间有限，很多细节都没有考虑，比如悔棋和心跳包的处理，在以后的继续学习中，应该会接触那些没有考虑到的细节。

实验 7：JDBC 方式操作数据库

- 实验题目

##实验 7：JDBC 方式操作数据库

###实现一个简单的信息管理系统

实现如下功能：

- 创建信息表；
- 实现记录的查询，修改，插入，删除；
- 实现记录的统计；
- 纪录数据的导入和导出（Excel 文件）

###例如

- 以下例子仅供参考，但必须要完成上述所列的基本功能。
- 可根据实际需要作适当的修改，补充。

####学生信息管理系统

- 创建 **student** 表，包含学生的学号、姓名、年龄信息。
- 根据学号，可以查询到学生的姓名和年龄；
- 给定学生的学号、姓名、年龄，在表中追加一行信息；
- 给定学生的学号，可以从表中删除该学生的信息；

####图书信息管理系统

- 创建图书信息表，包含图书的书名、书号、作者、出版日期信息。
- 根据书名，可以查询到图书的书号、作者、出版日期信息；
- 给定图书的书名、书号、作者、出版日期信息，在表中追加一行信息；
- 给定图书的书名，可以从表中删除该图书的信息；

####商品信息管理系统

- 创建商品信息表，包含商品的名称、规格、生产厂商、价格信息。
- 根据商品的名称，可以查询到商品的、规格、生产厂商、价格信息；
- 给定商品的名称、规格、生产厂商、价格信息，在表中追加一行信息；
- 给定商品的名称，可以从表中删除该商品的信息；

- 问题分析

需要解决的技术问题：数据库、Excel

采用的技术方案：mysql-connector-java、POI

*测试前需要将 mysql-connector 和 POI 的 jar 包导入 CLASSPATH（环境变量或命令行参数均可以）才能正常运行

- 实验的具体步骤

为了连接 mysql 数据库，使用了 mysql-connector-java 这个第三方库

为了操作 Excel，使用了 POI 这个第三方库（其中包含的 jar 很多）

- 实现程序的完整代码

```
import java.io.*;
import java.sql.*;
import java.util.Scanner;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;
import org.apache.poi.openxml4j.exceptions.InvalidFormatException;
import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.CellStyle;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.HorizontalAlignment;
import org.apache.poi.ss.usermodel.VerticalAlignment;
import org.apache.poi.ss.usermodel.BorderStyle;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.ss.usermodel.WorkbookFactory;

class StudentData {
    public int id;
    public String name;
    public int age;
}

interface GetStudent {
    public boolean student_callback(StudentData data);
}

class StudentExcel {
```

```

private Workbook wb;
private Sheet sheet;
private int num;

public StudentExcel() {
    wb = new XSSFWorkbook();
    sheet = wb.createSheet();
    num = 0;
}

public StudentExcel(String filename) throws IOException,
InvalidFormatException {
    wb = WorkbookFactory.create(new File(filename));
    sheet = wb.getSheetAt(0);
    num = sheet.getLastRowNum() + 1;
}

public StudentExcel insert(int id, String name, int age) {
    Row row = sheet.createRow(num++);
    row.setHeightInPoints(30);
    CellStyle cs = wb.createCellStyle();
    cs.setAlignment(HorizontalAlignment.CENTER);
    cs.setVerticalAlignment(VerticalAlignment.CENTER);
    cs.setBorderBottom(BorderStyle.THIN);
    cs.setBorderLeft(BorderStyle.THIN);
    cs.setBorderRight(BorderStyle.THIN);
    cs.setBorderTop(BorderStyle.THIN);
    Cell cell = row.createCell(0);
    cell.setCellStyle(cs);
    cell.setCellValue(id);
    cell = row.createCell(1);
    cell.setCellStyle(cs);
    cell.setCellValue(name);
    cell = row.createCell(2);
    cell.setCellStyle(cs);
    cell.setCellValue(age);
    return this;
}

public StudentExcel find(GetStudent gs) {
    boolean back = true;
    for(int i=0;back && i<num;i++) {
        Row row = sheet.getRow(i);
        StudentData data = new StudentData();
    }
}

```



```

        data.id = (int)row.getCell(0).getNumericCellValue();
        data.name = row.getCell(1).getStringCellValue().trim();
        data.age = (int)row.getCell(2).getNumericCellValue();
        back = gs.student_callback(data);
    }
    return this;
}

public StudentExcel writeTo(String filename) throws IOException {
    FileOutputStream fos = new FileOutputStream(filename);
    wb.write(fos);
    if(null != fos)
        fos.close();
    return this;
}
}

class StudentDb {

    private final String table = "student";
    private Connection conn;

    public StudentDb(Connection conn) {
        this.conn = conn;
    }

    public StudentDb insert(int id, String name, int age) {
        try {
            String sql = "INSERT INTO "+ table + " (id, name, age)
VALUE (?, ?, ?)";
            PreparedStatement preparedStmt =
conn.prepareStatement(sql);
            preparedStmt.setInt(1, id);
            preparedStmt.setString(2, name);
            preparedStmt.setInt(3, age);
            preparedStmt.execute();
            preparedStmt.close();
        } catch (SQLException e) {
            System.out.println("Can not insert data!");
        }
        return this;
    }

    public StudentDb delete(int id) {

```

```

        try {
            String sql = "DELETE FROM "+ table + " WHERE id = ?";
            PreparedStatement preparedStmt =
conn.prepareStatement(sql);
            preparedStmt.setInt(1, id);
            preparedStmt.execute();
            preparedStmt.close();
        } catch (SQLException e) {
            System.out.println("Can not delete data(id="+ id +")!");
        }
        return this;
    }

```

```

    public StudentDb modify(int id, String name, int age) {
        try {
            String sql = "UPDATE "+ table + " SET name=?, age=? WHERE
id=?";
            PreparedStatement preparedStmt =
conn.prepareStatement(sql);
            preparedStmt.setString(1, name);
            preparedStmt.setInt(2, age);
            preparedStmt.setInt(3, id);
            preparedStmt.execute();
            preparedStmt.close();
        } catch (SQLException e) {
            System.out.println("Can not modify data!");
        }
        return this;
    }

```

```

    public StudentData find(int id) {
        StudentData data = null;
        try {
            String sql = "SELECT * FROM "+ table + " WHERE id = ?";
            PreparedStatement preparedStmt =
conn.prepareStatement(sql);
            preparedStmt.setInt(1, id);
            ResultSet rs = preparedStmt.executeQuery();
            if (rs.next()) {
                data = new StudentData();
                data.id = rs.getInt("id");
                data.name = rs.getString("name");
                data.age = rs.getInt("age");
            }
        }
    }

```

```

        preparedStmt.close();
    } catch (SQLException e) {
        System.out.println("SQL Error");
    }
    return data;
}

public StudentDb find(GetStudent gs) {
    try {
        String sql = "SELECT * FROM " + table;
        Statement stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery(sql);
        boolean back = true;
        while (back && rs.next()) {
            StudentData data = new StudentData();
            data.id = rs.getInt("id");
            data.name = rs.getString("name");
            data.age = rs.getInt("age");
            back = gs.student_callback(data);
        }
        stmt.close();
    } catch (SQLException e) {
        System.out.println("SQL Error");
    }
    return this;
}

public StudentDb createTable() {
    try {
        Statement stmt = conn.createStatement();
        String sql = "CREATE TABLE IF NOT EXISTS " + table + " ( " +
            "id INT(11) PRIMARY KEY NOT NULL, " +
            "name TINYTEXT NOT NULL, " +
            "age INT(3) NOT NULL" +
            ") DEFAULT CHARSET=utf8";
        stmt.executeUpdate(sql);
        stmt.close();
    } catch (SQLException e){
        System.out.println("Can not create table!");
    }
    return this;
}

public StudentDb dropTable() {

```

```

        try {
            Statement stmt = conn.createStatement();
            String sql = "DROP TABLE IF EXISTS " + table;
            stmt.executeUpdate(sql);
            stmt.close();
        } catch (SQLException e){
            System.out.println("Can not create table!");
        }
        return this;
    }
}

public class JDBC {

    private static final String database = "java_exp";
    private static final String db_user = "java_exp";
    private static final String db_passwd = "ii0Yjvixn0Lif0SL";

    private static Connection conn;

    public static void close() {
        try {
            conn.close();
        } catch (SQLException e){
            System.out.println("Can not close the connection!");
            System.exit(-2);
        }
    }

    public static void main(String[] args) {
        try {
            PrintStream ps = new PrintStream(new
FileOutputStream("out/07-jdbc.txt"));
            System.setOut(ps);
        } catch (IOException e) {
            System.out.println("Failed to open file!");
            System.exit(-1);
        }

        String url="jdbc:mysql://localhost:3306/"+ database
+"?characterEncoding=utf8&useSSL=true";
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(url, db_user,

```

```

db_passwd);
    } catch(ClassNotFoundException e) {
        System.out.println("Can not find MySQL driver!");
        System.exit(-100);
    } catch (SQLException e){
        System.out.println("Can not connect the database!");
        System.exit(-101);
    }

    StudentDb stu_db = new StudentDb(conn);
    stu_db.dropTable();
    stu_db.createTable();

    try {
        StudentExcel se_in = new
StudentExcel("data/student.xlsx");
        se_in.find(new GetStudent() {
            public boolean student_callback(StudentData data) {
                stu_db.insert(data.id, data.name, data.age);
                return true;
            }
        });
    } catch (IOException e) {
        System.out.println("Failed to open file!");
        System.exit(-3);
    } catch (InvalidFormatException e) {
        System.out.println("Invalid Format Exception in the input
file!");
        System.exit(-4);
    }

    System.out.println("Original data\r\n-----");
    stu_db.find(new GetStudent() {
        public boolean student_callback(StudentData data) {
            System.out.println("id: " + data.id);
            System.out.println("name: " + data.name);
            System.out.println("age: " + data.age);
            System.out.println();
            stu_db.modify(data.id, data.name.toUpperCase(),
data.age + 1);
            if (data.id == 204)
                stu_db.delete(data.id);
            return true;
        }
    }

```

```

    });
    System.out.println("Data after modified\r\n-----
");
    stu_db.find(new GetStudent() {
        public boolean student_callback(StudentData data) {
            System.out.println("id: " + data.id);
            System.out.println("name: " + data.name);
            System.out.println("age: " + data.age);
            System.out.println();
            return true;
        }
    });
    System.out.println("Data of id = 203\r\n-----");
    StudentData data = stu_db.find(203);
    System.out.println("id: " + data.id);
    System.out.println("name: " + data.name);
    System.out.println("age: " + data.age);

    try {
        StudentExcel se_out = new StudentExcel();
        stu_db.find(new GetStudent() {
            public boolean student_callback(StudentData data) {
                se_out.insert(data.id, data.name, data.age);
                return true;
            }
        });
        se_out.writeTo("out/07-jdbc.xlsx");
    } catch (IOException e) {
        System.out.println("Failed to open file!");
    }
}
}

```

- 实验结果以及总结

结果:

out/07-jdbc.xlsx

| | A | B | C |
|---|-----|--------|----|
| 1 | 201 | DFAGTW | 13 |
| 2 | 202 | 张三 | 16 |
| 3 | 203 | 李四 | 21 |
| 4 | 205 | 铠甲 | 41 |
| 5 | 206 | 第三方 | 20 |
| 6 | 207 | FFEEDX | 24 |
| 7 | 208 | 董建华 | 22 |
| 8 | 209 | DD | 23 |

out/07-jdbc.txt

Original data

id: 201
name: dfagtw
age: 12

id: 202
name: 张三
age: 15

id: 203
name: 李四
age: 20

id: 204
name: 王五
age: 25

id: 205
name: 铠甲
age: 40

id: 206
name: 第三方

age: 19

id: 207

name: ffeedx

age: 23

id: 208

name: 董建华

age: 21

id: 209

name: dd

age: 22

Data after modified

id: 201

name: DFAGTW

age: 13

id: 202

name: 张三

age: 16

id: 203

name: 李四

age: 21

id: 205

name: 铠甲

age: 41

id: 206

name: 第三方

age: 20

id: 207

name: FFEEDX

age: 24

id: 208

name: 董建华

age: 22

id: 209
name: DD
age: 23

Data of id = 203

id: 203
name: 李四
age: 21

总结:

在这个实验中用到了两个第三方库，从这样的实验中，学会了导入并使用第三方库的方法，也明白了很多时候使用第三方库能大大简化工作量。

实验 8：网络编程基础

- 实验题目

##实验 8：网络编程基础

实现 Java 爬虫，抓去设定网站的信息

- 设定一家知名的信息网站，例如百度新闻，今日头条，新浪新闻，网易新闻等，
- 使用 `InetAddress` 类的方法获取设定网站的主机的 IP 地址；
- 获取本地机的名称和 IP 地址。
- 使用 `URL` 类下载首页信息，
- 分析首页信息，进行信息过滤，将新闻报道的正文进行分门别类的整理。

- 问题分析

需要解决的技术问题：`InetAddress` 类、`URL` 类、解析 html 文档

采用的技术方案：学习 `InetAddress` 类、`URL` 类，使用 `JSoup` 解析 html 文档

*需导入第三方库 `jsoup`

- 实验的具体步骤

是用文本编辑器完成以下代码

- 实现程序的完整代码

```
import java.io.*;
import java.net.*;
import java.util.List;
import java.util.ArrayList;
import org.jsoup.Jsoup;
import org.jsoup.nodes.Document;
import org.jsoup.nodes.Element;
import org.jsoup.select.Elements;

class News {
    public String title;
```

```

    public String url;

    public News(String title, String url) {
        this.title = title;
        this.url = url;
    }
}

class NewSpider {

    private Document doc;

    public NewSpider(String url) throws IOException {
        doc = Jsoup.connect(url).get();
    }

    public void parseNews(String selector, String part_name) {
        Elements news = doc.select(selector);
        System.out.println(part_name);
        System.out.println("-----");
        int skip = 0;
        for (int i=1;i<=news.size();++i) {
            Element e = news.get(i-1);
            String title = e.text().trim();
            String url = e.attr("href").trim();
            if (url.isEmpty()) {
                Elements urls = e.select("a[href]");
                url = urls.first().attr("href").trim();
                for (int j=1;j<urls.size();++j)
                    url += ", " + urls.get(j).attr("href").trim();
            }
            if (title.length() >= 6)
                System.out.println( (i-skip) + "\、" + title + "(" +
url + ")" );
            else
                skip ++;
        }
        System.out.println();
    }
}

public class Spider {

    // public static News[] parseNews(String html) {

```

```

// Document doc = Jsoup.parse(html);
// final String[] selectors = {
//     ".list_12>li>a",
//     ".list_14>li>a"
// };
// List<News> list = new ArrayList<News>();
// for (String selector : selectors) {
//     Elements news = doc.select(selector);
//     for (Element e : news) {
//         String title = e.text().trim();
//         String url = e.attr("href").trim();
//         if (title.length() >= 8)
//             list.add(new News(title, url));
//     }
// }
// News[] result = new News[list.size()];
// return list.toArray(result);
// }

public static String fetchHTML(String u) {
    StringBuffer content = new StringBuffer();
    try {
        URL url = new URL(u);
        HttpURLConnection con = (HttpURLConnection)
url.openConnection();
        con.setRequestMethod("GET");
        int status = con.getResponseCode();
        if (status != 200)
            return null;
        BufferedReader in = new BufferedReader(new
InputStreamReader(con.getInputStream(), "UTF-8"));
        String inputLine;
        while ((inputLine = in.readLine()) != null)
            content.append(inputLine + "\r\n");
        in.close();
        con.disconnect();
    } catch (ProtocolException e) {
        return null;
    } catch (IOException e) {
        return null;
    }
    return content.toString();
}

```

```

    public static void main(String[] args) {
        try {
            PrintStream ps = new PrintStream(new
FileOutputStream("out/08-spider.txt"), false, "UTF-8");
            System.setOut(ps);
        } catch (IOException e) {
            System.out.println("Failed to open file!");
            System.exit(-1);
        }

        try {
            InetAddress local = InetAddress.getLocalHost();
            InetAddress target =
InetAddress.getByName("news.sina.com.cn");
            System.out.println("Target Host Name: " +
target.getHostName());
            System.out.println("Target Host IP: " +
target.getHostAddress());
            System.out.println("Local Host Name: " +
local.getHostName());
            System.out.println("Local Host IP: " +
local.getHostAddress());
            System.out.println();
        } catch (UnknownHostException e) {
            System.out.println("Fail to resolve the host!");
        }

        // News[] all =
parseNews(fetchHTML("http://news.sina.com.cn/"));
        // for (News news : all) {
        // System.out.println(news.title + "(" + news.url + ")");
        // }

        try {
            NewSpider ns = new NewSpider("http://news.sina.com.cn/");
            ns.parseNews(".ct_t_01>h1", "大字要闻");
            ns.parseNews("#ad_entry_b2_b>.list_14>li>a[href]", "次级要
闻");
            ns.parseNews("#blk_08_cont01 a[href]", "军事");
            ns.parseNews("div[data-sudaclick=history_1] a[href]", "历
史");
            ns.parseNews("div[data-sudaclick=book_1] a[href]", "读书
");
            ns.parseNews("#blk_gnxw_011 a[href]", "内地新闻");

```

```
ns.parseNews("#blk_gjxw_011 a[href]", "国际新闻");
ns.parseNews("#blk_cjkjqcfc_011 a[href]", "财经·科技·汽车·房产·地产·教育 1");
ns.parseNews("#blk_cjkjqcfc_012 a[href]", "财经·科技·汽车·房产·地产·教育 2");
ns.parseNews("#blk_lctycp_011 a[href]", "娱乐·体育·彩票·游戏 1");
ns.parseNews("#blk_lctycp_012 a[href]", "娱乐·体育·彩票·游戏 2");
ns.parseNews("#blk_sh_011 a[href]", "社会 1");
ns.parseNews("#blk_sh_012 a[href]", "社会 2");
ns.parseNews("#blk_gntltop_01 + .part_02 a[href]", "国内新闻");
ns.parseNews("#blk_gjtltop_01 + .part_02 a[href]", "国际新闻");
ns.parseNews("#blk_jstltop_01 + .part_03 a[href]", "军事新闻");
ns.parseNews("#blk_jktltop_01 + .part_03 a[href]", "健康新闻");
ns.parseNews("#blk_tytltop_01 + .part_02 a[href]", "体育新闻");
ns.parseNews("#blk_cjtltop_01 + .part_02 a[href]", "财经新闻");
ns.parseNews("#blk_spsctltop_01 + .part_03 a[href]", "收藏");
ns.parseNews("#blk_fctltop_01 + .part_03 a[href]", "房产");
ns.parseNews("#blk_qctltop_01 + .part_03 a[href]", "汽车");
ns.parseNews("#blk_kjtltop_01 + .part_02 a[href]", "科技新闻");
ns.parseNews("#blk_bktltop_01 + .p_box a[href]", "博客");
ns.parseNews("#blk_dstltop_01 + .p_box a[href]", "读书");
ns.parseNews("#blk_jytltop_01 + .p_box a[href]", "教育 考试");
ns.parseNews("#blk_nxtltop_01 + .part_03 a[href]", "历史");
ns.parseNews(".newpart a[href]", "时尚 女性 星座");
ns.parseNews("#blk_yxtltop_01 + .part_03 a[href]", "游戏");
ns.parseNews("#blk_yltltop_01 + .part_02 a[href]", "娱乐新闻");
ns.parseNews("#blk_shtltop_01 + .part_04 a[href]", "社会新闻");
```

```
    } catch (IOException e) {  
        System.out.println("Can not load HTML!");  
    }  
}  
}
```

- 实验结果以及总结

实验结果:

输出文件过大（有 900 多行），这里只列出第一屏



```
1 Target Host Name: news.sina.com.cn  
2 Target Host IP: 58.205.212.209  
3 Local Host Name: LAPTOP-BHCQ6EN0  
4 Local Host IP: 192.168.153.1  
5  
6 大字要闻  
7 -----  
8 1、习近平就埃及严重恐袭向埃及总统致慰问电(http://news.sina.com.cn/china/xlwx/2017-11-25/doc-ifypapmz4968155.shtml)  
9 2、中央纪委官网曝光赵乐际的新身份(http://news.sina.com.cn/c/nd/2017-11-26/doc-ifypathz6037304.shtml)  
10 3、2人造谣红黄蓝幼儿园股东涉领导人亲属被查(http://news.sina.com.cn/c/2017-11-25/doc-ifypacti8176150.shtml)  
11 4、浙江宁波突发爆炸 30多人受伤(http://news.sina.com.cn/c/nd/2017-11-26/doc-ifypceiq2781618.shtml, http://news.sina.com.cn/c/nd/2017-11-26/doc-ifypceiq2831678.shtml)  
12 5、北京市安委会驳“驱赶低端人口”说(http://news.sina.com.cn/c/nd/2017-11-26/doc-ifypapmz5114368.shtml)  
13 6、习近平就埃及严重恐袭向埃及总统致慰问电(http://news.sina.com.cn/china/xlwx/2017-11-25/doc-ifypapmz4968155.shtml)  
14 7、北京警方通报红黄蓝幼儿园虐童事件(http://news.sina.com.cn/c/nd/2017-11-25/doc-ifypathz5911517.shtml)  
15 8、特朗普称拒《时代》年度人物提名 杂志回应(http://news.sina.com.cn/w/zx/2017-11-25/doc-ifypacti8109634.shtml)  
16 9、拿下首虎外 本周中纪委还释放个不一般信号(http://news.sina.com.cn/c/nd/2017-11-25/doc-ifypceiq1868061.shtml)  
17 10、十九大后 再有2省经常委跨省调整(图)(http://news.sina.com.cn/c/nd/2017-11-25/doc-ifypapmz4909243.shtml)  
18 11、金马奖:惠英红文淇“母女”包揽影后女配(http://ent.sina.com.cn/zt_d/goldenhorse54/)  
19  
20 次级要闻  
21 -----  
22 1、毛岸英牺牲67周年: 毛泽东曾珍藏其遗物26年(http://news.sina.com.cn/o/2017-11-25/doc-ifypapmz4913814.shtml)  
23 2、中朝友谊桥将临时关闭旨在对朝鲜施压? 中方回应(http://news.sina.com.cn/o/2017-11-24/doc-ifypapmz4723374.shtml)  
24 3、津巴布韦新总统上任中方是否发贺电? 外交部回应(http://news.sina.com.cn/o/2017-11-24/doc-ifypapmz4809636.shtml)  
25 4、禁止中国人入内? 日本顶级化妆品品牌被指歧视华人(http://news.sina.com.cn/w/zx/2017-11-25/doc-ifypathz5902348.shtml)  
26 5、刚跟中国“牵手”的这个国家 被称石油通道上的哨兵(http://news.sina.com.cn/c/nd/2017-11-24/doc-ifypapmz4812250.shtml)  
27 6、顺丰3.2亿在淘宝拍走两架波音747 外媒惊呆(图)(http://news.sina.com.cn/w/2017-11-25/doc-ifypacti7946719.shtml)  
28 7、王清亮任山东省委常委(图/简历)(http://news.sina.com.cn/c/nd/2017-11-25/doc-ifypceiq2375362.shtml)  
29 8、澳媒被指途中必反: 总理仅因饭局有中国人被狂骂(http://news.sina.com.cn/w/zx/2017-11-25/doc-ifypacti7960172.shtml)  
30 9、这个国家的总统卸任后要去孔子学院学汉语(图)(http://news.sina.com.cn/o/2017-11-25/doc-ifypacti8073612.shtml)  
31 10、23岁小伙被自己的车压死 因停车时少做一个动作(http://news.sina.com.cn/s/pa/2017-11-25/doc-ifypacti8007423.shtml)  
32 11、女童跑向正在关门的高铁 他的一个拥抱太暖心(图)(http://news.sina.com.cn/s/wh/2017-11-25/doc-ifypapmz4865336.shtml)  
33 12、处级警察举报科级检察官: 对证人刑讯逼供(http://news.sina.com.cn/c/nd/2017-11-24/doc-ifypacti7897908.shtml)  
34 13、女司机把别人撞成植物人 自己买车买房仅赔2.6万(http://news.sina.com.cn/s/wh/2017-11-25/doc-ifypceiq1692755.shtml)  
35 14、“美国马云”财富破千亿美元 因美人民都在干这事(http://news.sina.com.cn/o/2017-11-25/doc-ifypacti8017207.shtml)  
36 15、187类消费品进口关税税率降低(http://finance.sina.com.cn/china/gncj/2017-11-25/doc-ifypathz5770272.shtml)  
37 16、对你我影响几何? (http://finance.sina.com.cn/roll/2017-11-25/doc-ifypacti7962739.shtml)  
38 17、宗庆后食言谋划娃哈哈上市背后: 止不住的业绩下滑(http://finance.sina.com.cn/roll/2017-11-25/doc-ifypceiq1690766.shtml)
```

总结:

第一想法是使用正则表达式完成这道题,但很怀念之前用 jQuery 和 css 选择器快速获取 DOM 元素的便捷感,发现一神器——Jsoup 能实现类似的功能,遂使用了该第三方库完成本题。

实验 10：重复文件的查询

- 实验题目

。##实验 10：重复文件的查询

实现如下基本功能：

- 设定需要查询的文件目录；
- 在遍历整个文件目录的同时，获取每个文件的 MD5 checksum 值；
- 若在遍历工程之中若发现有的 MD5 checksum 值相同的文件时，将两个文件进行对比，确认两个文件是否相同；
- 若判断两个文件相同时，输出结果提示用户两个重复文件的路径以及文件名。

- 问题分析

需要解决的技术问题：

- 1.遍历整个文件目录及其子目录（递归）；
- 2.获取每个文件的 MD5 checksum 值；
- 3.合适数据结构（并查集）；
- 4.最终确认两个文件是否相同；

- 实验的具体步骤

用文本编辑器完成以下代码

- 实现程序的完整代码

```
import java.io.*;
import java.security.MessageDigest;
import java.security.DigestInputStream;
import java.security.NoSuchAlgorithmException;
import java.util.List;
import java.util.ArrayList;

class FileAttr {
    public String filename;
    public String md5;
    public int parent; // union find parent
```



```

public int size;

public static FileAttr[] list;

public static int find(int a) {
    if (list[a].parent == -1 || list[a].parent == a)
        return a;
    return list[a].parent = find(list[a].parent);
}

public static void union(int a, int b) {
    list[a].parent = a = find(a);
    list[b].parent = b = find(b);
    if (list[a].size > list[b].size) {
        list[b].parent = a;
        list[a].size += list[b].size;
    } else {
        list[a].parent = b;
        list[b].size += list[a].size;
    }
}

public FileAttr(String filename, String md5) {
    this.filename = filename;
    this.md5 = md5;
    parent = -1;
    size = 0;
}

public static void buildTree() {
    for (int i=0;i<list.length-1;++i) {
        for (int j=i+1;j<list.length;++j) {
            if (list[i].equals(list[j])) {
                union(i, j);
            }
        }
    }
    for (int i=0;i<list.length;++i) {
        if (list[i].parent != -1)
            find(i);
    }
}

public boolean equals(Object o) {

```

```

    if (o == this)
        return true;
    if (!(o instanceof FileAttr))
        return false;
    final FileAttr fa = (FileAttr)o;
    if (!fa.md5.equals(md5))
        return false;
    try {
        int bufferSize = 256 * 1024;
        FileInputStream f1 = new FileInputStream(filename);
        FileInputStream f2 = new FileInputStream(fa.filename);
        byte[] buffer1 = new byte[bufferSize];
        byte[] buffer2 = new byte[bufferSize];
        int n1, n2;
        do {
            n1 = f1.read(buffer1);
            n2 = f2.read(buffer2);
            if (n1 != n2)
                return false;
            for (int i=0;i<n1;++i) {
                if (buffer1[i] != buffer2[i])
                    return false;
            }
        } while (n1 > 0);
        f1.close();
        f2.close();
    } catch (IOException e) {
        return false;
    }
    return true;
}

public static void show() {
    for (int i=0;i<list.length-1;++i) {
        if (list[i].parent == -1)
            continue;
        System.out.println("md5 = " + list[i].md5);
        System.out.println("-----");
    };

    System.out.println(list[i].filename);
    for (int j=i+1;j<list.length;++j) {
        if (list[i].parent == list[j].parent) {
            System.out.println(list[j].filename);
            list[j].parent = -1;
        }
    }
}

```

```

        }
    }
    list[i].parent = -1;
    System.out.println();
}
}
}

```

```

public class FileDetect {

```

```

    public static String byteArrayToHex(byte[] byteArray) {
        char[] hexDigits = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
'A', 'B', 'C', 'D', 'E', 'F'};
        char[] resultCharArray = new char[byteArray.length * 2];
        int index = 0;
        for (byte b : byteArray) {
            resultCharArray[index++] = hexDigits[b >>> 4 & 0xf];
            resultCharArray[index++] = hexDigits[b & 0xf];
        }
        return new String(resultCharArray);
    }

```

```

    public static String md5_file(File file) throws IOException {
        int bufferSize = 256 * 1024;
        try {
            MessageDigest md = MessageDigest.getInstance("MD5");
            FileInputStream fis = new FileInputStream(file);
            DigestInputStream dis = new DigestInputStream(fis, md);
            byte[] buffer = new byte[bufferSize];
            while (dis.read(buffer) > 0);
            md = dis.getMessageDigest();
            byte[] resultByteArray = md.digest();
            dis.close();
            fis.close();
            return byteArrayToHex(resultByteArray);
        } catch (NoSuchAlgorithmException e) {
            return null;
        }
    }
}

```

```

    public static void getFileAttr(List<FileAttr> list, File dir)
throws IOException {
        File[] files = dir.listFiles();
        for (File f : files) {

```

```

        if (f.isFile())
            list.add(new FileAttr(f.getAbsolutePath(),
md5_file(f)));
        else if (f.isDirectory())
            getFileAttr(list, f);
    }
}

public static void main(String args[]) {
    try {
        PrintStream ps = new PrintStream(new
FileOutputStream("out/10-fileDetect.txt"));
        System.setOut(ps);
    } catch (IOException e) {
        System.out.println("Failed to open file!");
        System.exit(-1);
    }
    try {
        List<FileAttr> filelist = new ArrayList<FileAttr>();
        getFileAttr(filelist, new File("data/files"));
        FileAttr.list = filelist.toArray(new
FileAttr[filelist.size()]);
        FileAttr.buildTree();
        FileAttr.show();
    } catch (IOException e) {
        System.out.println("Can not read files!");
    }
}
}

```

- 实验结果以及总结

实验结果：

out/10-fileDetect.txt

md5 = 9482111E5BBB81CED49E805BAF7C9906

R:\java\experiments\data\files\1.txt

R:\java\experiments\data\files\6.txt

R:\java\experiments\data\files\subpath\3.txt

md5 = E30B8B9C97C05478798F535640BC302E

R:\java\experiments\data\files\2.txt

R:\java\experiments\data\files\subpath\5.txt

总结:

Md5 使用 Java 自带的库，另外递归、数据结构（这里用到了并查集）的思想也是十分重要的