

Uebung 5 Patryk Dajos

Aufgabe 1)

```
import java.util.Scanner;
import java.io.FileWriter;
import java.io.IOException;
public class Pruefziffer {
    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("Geben Sie Ihre Seriennummer ohne der Prüfziffer:");
        String userIn = input.nextLine();
        boolean test = true;
        long serialNr = 0;
        //Let's check the input first
        if (Character.isLetter(userIn.charAt(0))) {
            for (int i = 1; i < userIn.length(); i++) {
                if (!Character.isDigit(userIn.charAt(i))) {
                    System.out.println("Ihre Seriennummer ist nicht korrekt.");
                    test = false;
                }
            }
            if (test) {
                serialNr = ((int) Character.toLowerCase(userIn.charAt(0))) - 96;
                for (int j = 1; j < userIn.length(); j++) {
                    serialNr = serialNr * 10 + (Character.getNumericValue(userIn.charAt(j)));
                }
            }
        } else {
            System.out.println("Ihre Seriennummer ist nicht korrekt.");
            test = false;
        }
        long sum = calculateThis(serialNr);
        int result = 8 - ((int)sum % 9);
        if (result == 0) {
            result = 9;
        }
        System.out.println("Ihre Prüfziffer: " + result);
        userIn = userIn + "-" + result;
        input.close();
        try {
            FileWriter out = new FileWriter("'" + userIn + ".jpg");
            out.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

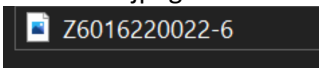
    public static long calculateThis(long number) {

        if (number <= 9)
            return number;

        return number % 10 + calculateThis(number / 10);
    }
}
```

Die Zahlen werden mit FileWriter gespeichert im src folder.

Leider nur jpegs aber die art wie diese zahlen abgespeichert werden wurde ja nicht erwähnt.



Geben Sie Ihre Seriennummer ohne der Prüfziffer:
Z6016220022
Ihre Prüfziffer: 6

Aufgabe 2)

1)

```
int[] kek = new int[2];
```

2)

```
kek[0] = 1;
```

3)

```
System.out.println(kek[0]);
```

4)

Weil diese methode ein void ist.

Aufgabe 3)

1)

```
1 import java.util.Arrays;
2 import java.util.Scanner;
3 public class Array {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7         System.out.println("Wie lang soll Ihr Array werden?");
8         int length = input.nextInt();
9         System.out.println("Geben Sie jetzt Ihre Zahlen ein: ");
10        int[] numbers = new int[length];
11        for (int i = 0; i < length; i++) {
12            numbers[i] = input.nextInt();
13        }
14        Arrays.sort(numbers);
15        System.out.println("Ihre kleinste Zahl: " + numbers[0]);
16        System.out.println("Ihre kleinste Zahl: " + numbers[length - 1]);
17        System.out.println("Ihr Array enthält: ");
18        int j = 1;
19        while (j < length - 1) {
20            if (numbers[j] != numbers[j - 1]) {
21                System.out.println(numbers[j]);
22            }
23            j++;
24        }
25        input.close();
26    }
27 }
```

```

Wie lang soll Ihr Array werden?
8
Geben Sie jetzt Ihre Zahlen ein:
1
1
2
2
1
7
5
8
Ihre kleinste Zahl: 1
Ihre kleinste Zahl: 8
Ihr Array enthält:
2
5
7

```

2)

```

public class Wuerfel {
    Run | Debug
    public static void main(String[] args) {
        int[] wuerfel = new int[] { 0, 0, 0, 0, 0, 0 };
        boolean goOn = true;
        while (goOn) {
            int temp = (int) (Math.random() * ((5) + 1));
            wuerfel[temp] = wuerfel[temp] + 1;
            if (wuerfel[temp] == 1000000) {
                goOn = false;
            }
        }
        System.out.println("Häufigkeiten:");
        System.out.println("1: " + wuerfel[0] + "\t" + "2: " + wuerfel[1] + "\t" + "3: " + wuerfel[2] + "\t" + "4: "
            + wuerfel[3] + "\t" + "5: " + wuerfel[4] + "\t" + "6: " + wuerfel[5]);
    }
}

```

```

Häufigkeiten:
1: 998294      2: 998301      3: 997743      4: 1000000      5: 998092      6: 999785
PS D:\Important\School\PP\Sem1\Uebung 5\uebung5> 

```

Aufgabe 4)

```

import java.util.Scanner;

public class Sieb {
    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("Geben Sie Ihre Schranke an: ");
        int length = input.nextInt();
        boolean[] menge = new boolean[length];
        for (int i = 3; i < length; i++) {
            boolean test = true;
            for (int j = i - 1; j > 1; j--) {
                if (i % j == 0) {
                    test = false;
                }
            }
            for (int j = i + 1; j < length; j++) {
                if (i % j == 0) {
                    test = false;
                }
            }
            if (test) {
                menge[i] = true;
            }
        }
        System.out.println("Ihre Primzahlen: ");
        for (int i = 3; i < length; i++) {
            if (menge[i]) {
                System.out.print(i + " | ");
            }
        }
        input.close();
    }
}

```

Geben Sie Ihre Schranke an:

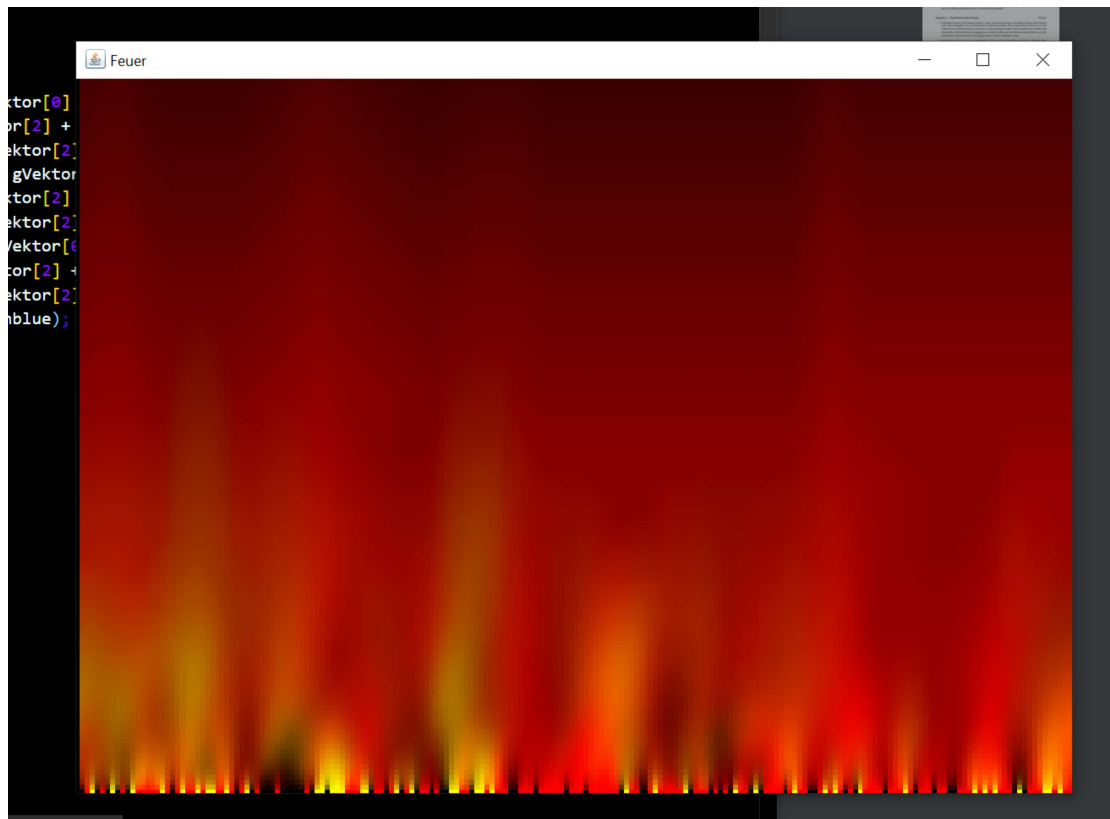
50

Ihre Primzahlen:

3 | 5 | 7 | 11 | 13 | 17 | 19 | 23 | 29 | 31 | 37 | 41 | 43 | 47 |

Das auslesen wurde komplizierter als ich es mir gedacht habe löf

Aufgabe 5)



```

while (true) {
    int i = 1;
    while (i != 8) {
        colors[(int) (Math.random() * WIDTH)][1] = Color.YELLOW;
        i++;
    }
    i = 1;
    while (i != 16) {
        colors[(int) (Math.random() * WIDTH)][1] = Color.RED;
        i++;
    }
    i = 1;
    while (i != 8) {
        colors[(int) (Math.random() * WIDTH)][1] = Color.BLACK;
        i++;
    }

    int[] gVektor = new int[] { 1, 1, 1, 3 };
    int x = 1;
    int y = 2;
    while (y < HEIGHT) {
        x = 1;
        while (x < WIDTH) {
            int l = x - 1;
            if (l == -1 || l == 0) {
                l = x+1;
            }
            int r = x + 1;
            if (r >= WIDTH) {
                r = x-1;
            }

            int nred = (colors[x][y].getRed() * gVektor[0] + colors[l][y].getRed() * gVektor[1]
                + colors[r][y].getRed() * gVektor[2] + colors[x][y - 1].getRed() * gVektor[3])
                / (gVektor[0] + gVektor[1] + gVektor[2] + gVektor[3]);
            int ngreen = (colors[x][y].getGreen() * gVektor[0] + colors[l][y].getGreen() * gVektor[1]
                + colors[r][y].getGreen() * gVektor[2] + colors[x][y - 1].getGreen() * gVektor[3])
                / (gVektor[0] + gVektor[1] + gVektor[2] + gVektor[3]);
            int nblue = (colors[x][y].getBlue() * gVektor[0] + colors[l][y].getBlue() * gVektor[1]
                + colors[r][y].getBlue() * gVektor[2] + colors[x][y - 1].getBlue() * gVektor[3])
                / (gVektor[0] + gVektor[1] + gVektor[2] + gVektor[3]);
            colors[x][y] = new Color(nred, ngreen, nblue);
            x++;
        }
        y++;
    }
    ImageFrame.drawArray(colors);

    try {
        Thread.sleep(50);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

```

