
河南工业大学

《JAVA 程序设计》实验报告

专业班级： 物联网 1603 学号： 201616070320 姓名： 郭治洪

实验单元一 Java 基本语法

实验一 JDK 安装、配置及 Java 程序的编译和运行

实验时间： 2018.9

【实验目的】

- 1、熟悉 JDK 的安装、配置。
- 2、学会如何编辑、编译、运行 Java 程序。

【实验环境】

JDK

【实验内容】

1. 编写一个简单的程序，输出“Welcome to java world”。
2. 编程求解 234 是否是一个水仙花数。“水仙花数”是指一个 3 位数，其各位数字立方和等于该数。

【详细分析】

实验 1 没什么难度，实验 2 无非就是一个三位数的分离，使用除法和取余就可以做到。

不过我看到在 JDK10 中，可以用 var 定义变量，让编译器进行自动局部变量推断，特性：**必须是有效值**。

【实验源码】

1. 实验 1 源码

```
/*
 * Java Experiment 01:Hello Java
 * Data:2018.9
 * Collage:Internet of Things
 * Class:IoT 1603
 * Name:GuoZhiHong
```

```
* Student ID:201616070320
*/

package Experiment01;

public class Experiment01 {

    public static void main(String[] args) {
        System.out.println("Welcome to java world");
    }
}
```

2. 实验 2 源码

```
/*
 * Java Experiment 02:Daffodil number
 * Data:2018.9
 * Collage:Internet of Things
 * Class:IoT 1603
 * Name:GuoZhiHong
 * Student ID:201616070320
 */

package Experiment02;

public class Experiment02 {

    public static void main(String[] args) {
        int testnum1=234;
        var testnum2=234; /*
                                This is new characteristic in JDK10
                                It can do the Local-Variable Type Inference.
                                See here:
                                JEP 286: Local-Variable Type
                                Inference<http://openjdk.java.net/jeps/286>
                                Chinese
                                version:https://segmentfault.com/a/1190000013612633
                                */

        int singledigit1=testnum1%10;
        var singledigit2=testnum2%10;
        int ten1=testnum1%100/10;
        var ten2=testnum1%100/10;
        int hundreds1=testnum1/100;
        var hundreds2=testnum1/100;
        boolean reslut1,reslut2;
```

```

        reslut1=(testnum1==singledigit1*singledigit1*singledigit1+ten1*ten1*ten
1+hundreds1*hundreds1*hundreds1);
        System.out.println("Test 1 reslut is "+reslut1);

        reslut2=(testnum2==singledigit2*singledigit2*singledigit2+ten2*ten2*ten
2+hundreds2*hundreds2*hundreds2);
        System.out.println("Test 2 reslut is "+reslut2);
    }
}

```

【实验结果】

1. 实验 1 截图

The screenshot shows an IDE with four tabs: Homework.java, Experiment01.java, Experiment02.java, and Experiment03.java. The Experiment01.java tab is active, displaying the following code:

```

1
2 /*
3  * Java Experiment 01:Hello Java
4  * Data:2018.9
5  * Collage:Internet of Things
6  * Class:IoT 1603
7  * Name:GuoZhiHong
8  * Student ID:201616070320
9  */
10
11 package Experiment01;
12
13 public class Experiment01 {
14
15     public static void main(String[] args) {
16         System.out.println("Welcome to java world");
17     }
18 }
19

```

At the bottom, the Console tab is active, showing the output:

```

<terminated> Experiment01 (1) [Java Application] D:\java\bin\javaw.exe (2018年9月11日 上午11:20:11)
Welcome to java world

```

2. 实验 2 截图

在下页。

```
Homework.java Experiment01.java Experiment02.java Experiment03.java
1  /*
2   * Java Experiment 02:Daffodil number
3   * Data:2018.9
4   * Collage:Internet of Things
5   * Class:IoT 1603
6   * Name:GuoZhiHong
7   * Student ID:201616070320
8   */
9
10 package Experiment02;
11
12 public class Experiment02 {
13
14     public static void main(String[] args) {
15         int testnum1=234;
16         var testnum2=234; /*
17
18             This is new characteristic in JDK10
19             It can do the Local-Variable Type Inference.
20             See here:
21             JEP 286: Local-Variable Type Inference<http://openjdk.java.net/jeps/286>
22             Chinese version:https://segmentfault.com/a/1190000013612633
23         */
24         int singledigit1=testnum1%10;
25         var singledigit2=testnum2%10;
26         int ten1=testnum1%100/10;
27         var ten2=testnum1%100/10;
28         int hundreds1=testnum1/100;
29         var hundreds2=testnum1/100;
30         boolean reslut1,reslut2;
31         reslut1=(testnum1==singledigit1*singledigit1*singledigit1+ten1*ten1*ten1+hundreds1*hundreds1*hundreds1
32         System.out.println("Test 1 reslut is "+reslut1);
33         reslut2=(testnum2==singledigit2*singledigit2*singledigit2+ten2*ten2*ten2+hundreds2*hundreds2*hundreds2
34
35     }
36 }
```

Problems @ Javadoc Declaration Console

<terminated> Experiment02 (1) [Java Application] D:\java\bin\javaw.exe (2018年9月11日 上午11:23:53)

Test 1 reslut is false
Test 2 reslut is false

【实验体会】

我觉得应该跟随新的标准新的要求，使用新的语法，了解新的知识，就像知道 JDK10 可以用自动局部变量推断一样。

另外自己动手做，自己查资料才最棒。

Go on!

实验二 Java 基本语法

实验时间： 2018.9

【实验目的】

- 1、熟悉 Java 语言中的数据类型、变量声明、数组、运算符号、流程控制语句。
- 2、学会定义类和方法，利用方法传递参数，得到方法的返回值。

【实验环境】

JDK, Eclipse

【实验内容】

- 1、计算两个 3*3 矩阵相乘，矩阵为 int[][]或者 double[][]都可以。声明一个方法，该方法接受参数，并返回计算的结果。
- 2、声明一个类，定义一个方法以计算一维数组中的最大值并返回该值，参数为 int[]或 double[]。在 main 方法中调用该方法，传递不同长度的数组，得到返回值并输出

【详细分析】

数学分析：

矩阵相乘：

a11	a12	a13	*	b11	b12	b13	=	a11*b11+a12*b21+a13*b31	a11*b12+a12*b22+a13*b32	a11*b13+a12*b23+a13*b33
a21	a22	a23		b21	b22	b23		a21*b11+a22*b21+a32*b31	a21*b12+a22*b22+a23*b32	a21*b13+a22*b23+a23*b33
a31	a32	a33		b31	b32	b33		a31*b11+a32*b21+a33*b31	a31*b12+a32*b22+a33*b32	a31*b13+a32*b23+a33*b33

矩阵 A

矩阵 B

结果矩阵 C

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}$$

数组索引下标从 0 开始，数组长度-1=数组下标!!

【实验源码】

1. 实验 1

```
/*
 * Java Experiment 03:Matrix multiplication
 * Data:2018.9
 * Collage:Internet of Things
 * Class:IoT 1603
 * Name:GuoZhiHong
 * Student ID:201616070320
 */

package Experiment03;

import java.util.Scanner;

public class Experiment03 {

    public static void main(String[] args) {
        double[][] a=new double[3][3];
        double[][] b=new double[3][3];

        Scanner in1 = new Scanner(System.in);
        for (int i = 0; i < a[0].length; i++) {
            for (int j = 0; j < a.length; j++) {
                System.out.println("Input martix A
["+i+"]["+j+"]:");
                a[i][j] = in1.nextInt();
            }
        }

        Scanner in2 = new Scanner(System.in);
        for (int i = 0; i < b[0].length; i++) {
            for (int j = 0; j < b.length; j++) {
                System.out.println("Input martix B
["+i+"]["+j+"]:");
                b[i][j] = in2.nextInt();
            }
        }

        double[][] reslut=MatrixMultiplication(a,b);

        for(int i=0;i<reslut[0].length;i++) { //Print reslut matrix
```

```

        for(int j=0;j<reslut.length;j++) {
            System.out.print(" "+reslut[i][j]+" ");
        }
        System.out.print("\n");
    }
}

public static double[][] MatrixMultiplication(double[][]
a,double[][] b) {

    if(a.length!=b[0].length) {//if they have different matrix
dimensions
        System.out.println("Input Matrixs have different matrix
dimensions!");
        return null;
    }

    //else
    double[][] reslut=new double[a[0].length][b.length];
    for(int i=0;i<a[0].length;i++) {
        for(int j=0;j<b.length;j++) {
            reslut[i][j]=0;
            for(int k=0;k<a.length;k++) {
                //for(int k=0;k<b[0].length;k++) {
                reslut[i][j]+=a[i][k]*b[k][j];
            }
        }
    }
    return reslut;
}
}

```

2. 实验 2

```

/*
 * Java Experiment 04:Find the maximum value in the array
 * Data:2018.9
 * Collage:Internet of Things
 * Class:IoT 1603
 * Name:GuoZhiHong
 * Student ID:201616070320
 */

package Experiment04;

```

```

import java.util.Scanner;

public class Experiment04 {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);
        double[] array=new double[10];
        for(int i=0;i<array.length;i++) {
            System.out.println("Input array A ["+i+"]");
            array[i]=in.nextInt();
        }
        System.out.println("The Method 1 's reslut is
"+FindTheMaximumValueInTheArrayMethod1(array));
        System.out.println("The Method 2 's reslut is
"+FindTheMaximumValueInTheArrayMethod2(array));
    }

    public static double
FindTheMaximumValueInTheArrayMethod1(double[] array) {
        double reslut=array[0];//set the initia value
        for (double i : array) { //for every value in the array
            if(i>reslut) //reset reslut
                reslut=i;
        }
        return reslut;
    }
    public static double
FindTheMaximumValueInTheArrayMethod2(double[] array) {
        double reslut=array[0];//set the initia value
        for (int index=array.length-1;index>=2;index--) { //for
every value in the array
            //index=length-1
            if(array[index]>reslut) //reset reslut
                reslut=array[index];
        }
        return reslut;
    }
}

```

【实验结果】

1. 实验 1


```
Window Help
Homework.java Experiment01.java Experiment02.java Experiment03.java
42      System.out.print("\n");
43  }
44  }
45  public static double[][] MatrixMultiplication(double[][] a, double[][] b) {
46
47      if(a.length!=b[0].length) { //if they have different matrix dimensions
48          System.out.println("Input Matrixs have different matrix dimensions!");
49          return null;
50      }
51
52      //else
53      double[][] reslut=new double[a[0].length][b.length];
54      for(int i=0;i<a[0].length;i++) {
55          for(int j=0;j<b.length;j++) {
56              reslut[i][j]=0;
57              for(int k=0;k<a.length;k++) {
58                  //for(int k=0;k<b[0].length;k++) {
59                      reslut[i][j]+=a[i][k]*b[k][j];
60              }
61          }
62      }
63      return reslut;
64  }
65
66  }
67
<terminated> Experiment03 [Java Application] D:\java\bin\javaw.exe (2018年9月11日 下午7:51:58)
1
6.0 6.0 6.0
15.0 15.0 15.0
24.0 24.0 24.0
```

2. 实验 2

```
Help
Experiment04.java Homework.java Experiment03.java
12
13 public class Experiment04 {
14
15  public static void main(String[] args) {
16
17      Scanner in = new Scanner(System.in);
18      double[] array=new double[10];
19      for(int i=0;i<array.length;i++) {
20          System.out.println("Input array A ["+i+"]");
21          array[i]=in.nextInt();
22      }
23      System.out.println("The Method 1 's reslut is "+FindTheMaximumValueInTheArrayMethod1(array));
24      System.out.println("The Method 2 's reslut is "+FindTheMaximumValueInTheArrayMethod2(array));
25  }
26
27
28  public static double FindTheMaximumValueInTheArrayMethod1(double[] array) {
29      double reslut=array[0]; //set the initia value
30      for (double i : array) { //for every value in the array
31          if(i>reslut) //reset reslut
32              reslut=i;
33      }
34      return reslut;
35  }
36  public static double FindTheMaximumValueInTheArrayMethod2(double[] array) {
37      double reslut=array[0]; //set the initia value
38      for (int index=array.length-1; index>=2; index--) { //for every value in the array
39          //index=length-1
40          if(array[index]>reslut) //reset reslut
41              reslut=array[index];
42      }
43      return reslut;
44  }
45  }
46  }
47
<terminated> Experiment04 [Java Application] D:\java\bin\javaw.exe (2018年9月11日 下午8:35:00)
Input array A [8]
9
Input array A [9]
10
The Method 1 's reslut is 10.0
The Method 2 's reslut is 10.0
```

【实验体会】

实验二有些花费时间，因为我的数学知识不过关，所以想了很久，不过在认真分析后还是做了出来。

另外在实验中遇到 **BUG**，我错误的把数组长度当成索引下标了（正确是数组长度-1=索引下标），然后排查发现了，下次要注意。

所以我觉得我应该好好学数学，多练习，多加小心，认真点。