电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2017221305016

姓 名 孙一淼

（实验） 课程名称 面向对象程序设计（Java）

理论教师 周帆

实验教师 何中海

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：孙一淼 学号：2017221305016 指导教师：何中海**

**实验地点：信软楼304 实验时间：2018.12.9**

1. **实验名称：JDK使用及Java基础**
2. **实验学时：2课时**
3. **实验目的：**熟悉JDK安装和参数方法；

熟悉Java的基本结构，变量，数组，数据类型。

1. **实验原理：**

1.通过对JDK1.8开发工具安装；

2.系统环境配置或设置。掌握开发工具的使用方法；

3.使用文字编辑工具编写程序；

1. **实验内容：**

项目1.完成第一章习题5、6编程。

项目2.完成第二章习题7编程。

项目3.编写程序，计算一个整数的各位数字之和，例如，整数20180925，

则计算并显示2+0+1+8+0+9+2+5的值。

项目4.打印输出斐波拉契数列。

1. **实验器材（设备、元器件）：**

PC电脑一台，Java教材一本。

1. **实验步骤：**

①安装JDK，配置系统环境变量。

②尝试运用记事本编写java文件，并用cmd进行编译和运行。

③运用IDE（Eclipse）进行java程序编写，项目、包和类的创建。

④编写习题，计算某一个整数的各位数之和，打印斐波拉契数列。

⑤总结与反思，寻找更优的解决方案。

1. **实验结果与分析（含重要数据结果分析或核心代码流程分析）**

项目1：第一章习题5、6编程

|  |
| --- |
| **实验步骤：**  ①将核心代码复制至Editplus中，另存为Experiment.java文件，并将编码方式改变为“UTF-8”（无BOM）；  ②在cmd中将当前目录切换至Experiment.java的目录中，先进行编译（javac Experiment.java），然后再进行运行，最终得到输出。  **核心代码如下：**  **public** **class** Experiment {  **public** **static** **void** main(String[] args) {  System.***out***.println("Welcome to Java World!");  }  } |
| **测试结果1：**  C:\Users\三水大大\AppData\Roaming\Tencent\Users\826301481\QQ\WinTemp\RichOle\7V]JYE[(SIBWF0}PJ0A0})8.png |
| **实验结论**：运用windows自带的记事本，其编码方式为“UTF-8”（+BOM），这样的编码方式java会报错，所以运用Editplus，采用“UTF-8”（无BOM）的编码方式，能够顺利的编译并运行java程序，最终得到正确的输出。 |

项目2：第二章习题7编程

|  |
| --- |
| **核心代码如下：**  **public** **class** Multiply {  **public** **static** **void** main(String[] args) {  **int** i, j;  **for** (i = 1; i <= 9; i++) {  **for** (j = 1; j <= i; j++) {  System.***out***.print(j + "\*" + i + "=" + i \* j + '\t');  //运用System.out.print输出不换行，用制表符将相邻乘法隔开  }  System.***out***.println();  //用System.out.print输出换行，不同数的乘法就换行  }  }  } |
| **测试结果1：**  C:\Users\三水大大\AppData\Roaming\Tencent\Users\826301481\QQ\WinTemp\RichOle\H{]0(}Q$0A6X@~Z0TYJ5A{2.png |
| **实验结论**：运用两重循环，以System.out.print输出不换行，输出对同一个数的乘法；以System.out.println输出换行，输出对不同数的乘法，最终输出正确的9\*9乘法表，实验结果正确。 |

项目3.编写程序，计算一个整数的各位数字之和

|  |
| --- |
| **核心代码如下：**  **import** java.util.Scanner;  **public** **class** AddNumber {  **public** **static** **void** main(String[] args) {  System.***out***.print("请输入一个数：");  Scanner in = **new** Scanner(System.***in***);  **long** number = in.nextLong();  // 可能输入的数比较长，所以用long型  **long** sum = 0;  **while** (number > 0) {  sum = sum + (number % 10);  // 模10运算得到各个位置上的数  number /= 10;  // 除以10取整  }  System.***out***.print("该数字各位数之和为：" + sum);  }  } |
| **测试结果1：**  C:\Users\三水大大\AppData\Roaming\Tencent\Users\826301481\QQ\WinTemp\RichOle\DH(9N_[6%9VTU){M9~PT6PG.png |
| **实验结论**：各位上的数字可以通过原数字从左往右依次模10运算得到，然后用一个变量sum将每次模运算得到的数字加起来，就得到了该数字所有位数相加之和。最后输入一个数字20180925，输出各位相加之和27，实验结果正确。 |

项目4.打印输出斐波拉契数列。

|  |
| --- |
| **核心代码如下：**  **Version1:**  **package** Fibonacci;  **public** **class** Fibonacci1 {  **public** **static** **void** main(String[] args) {  **int** i = 1;  **int** j = 1;  **int** n = 1;// n为计数器  **int** tmp = 0;  // tmp为下一个需要输出的数字  System.***out***.println(i);  System.***out***.println(j);  **for** (; n <= 15; n++) {  tmp = i + j;  // 后一位数字为i+j  System.***out***.println(tmp);  i = j;  j = tmp;  // j指向当前数字，i指向前一个数字  }  }  }  **Version2：(简化计算)**  **package** Fibonacci;  **public** **class** Fibonacci2 {  **public** **static** **void** main(String[] args) {  **int** i = 1;  **int** j = 1;  **int** n = 1;// n为计数器  **for** (; n <= 15; n++) {  System.***out***.println(i);  System.***out***.println(j);  i = i + j;  j = i + j;  }  }  }  **Version3:（运用数列）**  **package** Fibonacci;  **public** **class** Fibonacci3 {  **public** **static** **void** main(String[] args) {  **int** a[] = **new** **int**[15];  **int** i;  a[0] = 1;  a[1] = 1;  // 先定义数组开头的两个元素  **for** (i = 0; i < a.length; i++) {  **if** (i > 1) {  a[i] = a[i - 1] + a[i - 2];  // 如果a[i]为第三个数，则给第三个数赋值为前两个之和  }  System.***out***.println(a[i]);  }  }  }  **Version4:（调用函数）**  **package** Fibonacci;  **public** **class** Fibonacci4 {  **public** **static** **int** Fibonacci(**int** number) {  **if** (number == 1 || number == 2) {  **return** 1;  // 如果是数列的第一或者第二个元素，则返回1  } **else** {  **return** (*Fibonacci*(number - 1) + *Fibonacci*(number - 2));  } // 如果是第三个及之后的元素，则返回前两个元素之和  }  // 主函数  **public** **static** **void** main(String[] args) {  **int** i;  **for** (i = 1; i < 15; i++) {  System.***out***.println(*Fibonacci*(i));  // 调用函数打印前十五个斐波拉契数列中的元素  }  }  } |
| **测试结果1：**  C:\Users\三水大大\AppData\Roaming\Tencent\Users\826301481\QQ\WinTemp\RichOle\54D$)E2A7D}M_%92XYOFV59.png  **测试结果2：**  C:\Users\三水大大\AppData\Roaming\Tencent\Users\826301481\QQ\WinTemp\RichOle\@WG}%JZ[B5SM_(7RXZAFKIX.png |
| **实验结论**：可以使用多种方法打印斐波拉契数列，可以设置临时变量存放第三个数，也可以利用算法，不需设置第三个数的临时变量，还可以运用数列和递归函数进行运算，最终成功打印出斐波拉契数列，实验结果正确。 |

1. **总结及心得体会：**

|  |
| --- |
| ①工欲善其事必先利其器，在编写程序之前一定先配置好环境变量，否则可能会导致一系列报错，致使正确的代码无法通过编译，成功运行。  ②Java语言的许多语法与C类似，算法思想也有相同之处，入门相对比较容易，但是需要注意的细节地方很多，比如说代码中类的名称要和class文件的名称相同等等。  ③对于同一个程序可能有不同的实现方法，比如斐波拉契数列，就有四种方式可以实现，多尝试不同的方法，不仅可以锻炼我们的思考能力，还能为我们在实际工程中选取更优的方法起到很大的帮助。  ④在程序实现的过程中可能会遇到各种各样的问题，但是我们要坚持不懈地去发现问题所在并解决它，比如说用javac 编译文本文件的时候报错，我尝试了检查代码、更换编码方式（ANSI等等）都没成功，最后通过下载Editplus，更换无BOM的编码方式，最终成功解决。解决问题的过程不是浪费时间，也是我们积累经验和学习的好机会。 |

1. **对本实验过程及方法、手段的改进建议：**

|  |
| --- |
| ①在使用JDK软件包工具编译welcome to java world程序时，建议使用Editplus以及Notepad++等文本编辑器，并更改编码方式为“UTF-8”(无BOM)。  ②打印斐波拉契数列建议使用多种方法，临时变量存取、数组存放以及函数调用等各种方法均可尝试。 |

**报告评分：**

**指导教师签字：**