

电子科技大学信息与软件工程学院

实 验 报 告

(二)

学 号 2018091618008

姓 名 袁昊男

(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：袁昊男 学号：2018091618008 指导教师：周帆

实验地点：信软楼 304

实验时间：2019. 11. 24、12. 01

一、 实验名称：类和对象

二、 实验学时：2 学时

三、 实验目的：

- 1、熟悉 Java 的类、对象的基本操作；
- 2、理解面向对象程序设计的基本思想。

四、 实验原理：

1、 面向对象

Java 语言具有真正的面向对象语言的特点，除了数值、布尔和字符三种基本的数据类型外，其它类型都是对象。它支持封装、多态性和继承。封装就是将对象内的数据和代码联编起来，形成一个对象；多态性是指一个接口，有多个内在实现形式表示；继承是指某一对象直接使用另一对象的所有属性和方法的过程。

对程序员来说，这意味着要注意应中的数据 and 操纵数据的方法（method），而不是严格地用过程来思考。在一个面向对象的系统中，类（class）是数据和操作数据的方法的集合。数据和方法一起描述对象（object）的状态和行为。每一对象是其状态和行为的封装。类是按一定体系和层次安排的，使得子类可以从超类继承行为。在这个类层次体系中有一个根类，它是具有一般行为的类。Java 程序是用类来组织的。

2、 编译和解释性

Java 编译程序生成字节码（byte-code），而不是通常的机器码。Java 字节码提供对体系结构中性的目标文件格式，代码设计成可有效地传送程序到多个平台。Java 程序可以在任何实现了 Java 解释程序和运行系统（run-time system）的系统上运行。

在一个解释性的环境中，程序开发的标准“链接”阶段大大消失了。如果说 Java 还有一个链接阶段，它只是把新类装进环境的过程，

它是增量式的、轻量级的过程。因此, Java 支持快速原型和容易试验, 它将导致快速程序开发。这是一个与传统的、耗时的“编译、链接和测试”形成鲜明对比的精巧的开发过程。

五、 实验内容:

- 1、完成第三章习题 8 编程;
- 2、完成第四章习题 9、10、11 编程;
- 3、实现两个类, 分别是长方形和三角形求面积;
- 4、编写程序, 找出 $1\sim n$ 以内的所有素数。要求使用数组元素的下标从 $1\sim n$ 以内表示这些数值, 数组元素的值作为素数的标志。其中用数组元素的值为 0 来表示该元素的下标的数值是素数, 用 1 来表示该元素的下标的数值不是素数, 并输出这些素数。

六、 实验器材 (设备、元器件):

配置了 JDK 环境、安装有 Eclipse 软件的个人电脑一台。

七、 实验步骤:

- 1、按照题目要求分析需求、功能;
- 2、设计算法, 编写程序并进行测试;
- 3、结果分析, 撰写实验报告。

八、 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)

- 1、第三章习题 8:

编写一个类 TestArray, 它只有一个 main() 方法, 在该方法中, 创建一个 int 类型的一位数组 sim, 实现将数组 sim 的元素从小到大排序, 并输出排序后数组的值。

- (1) 代码

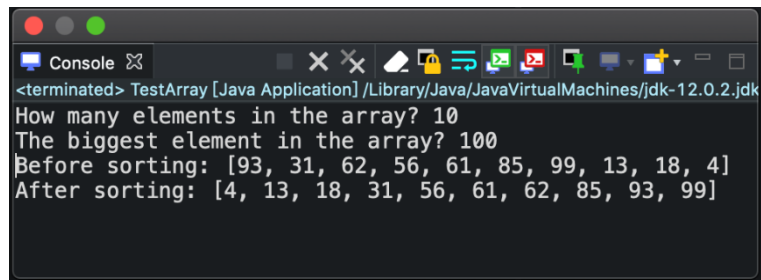
```
1. package pkg;
2.
3. import java.util.*;
4.
5. public class TestArray {
6.     public static void main(String[] args) {
7.         Scanner in = new Scanner(System.in);
8.
9.         System.out.print("How many ele-
10.             ments in the array? ");
11.         int n = in.nextInt();
12.         System.out.print("The biggest ele-
13.             ment in the array? ");
14.         int max = in.nextInt();
15.         int[] array = new int[n];
```

```

14.
15.     for (int i = 0; i < array.length; i++) {
16.         int e = (int) (Math.random() * max);
17.         array[i] = e;
18.     }
19.
20.     System.out.println("Before sorting: " + Ar-
    rays.toString(array));
21.     Arrays.sort(array);
22.
23.     in.close();
24.     System.out.println("After sorting: " + Ar-
    rays.toString(array));
25. }
26. }

```

(2) 运行截图



```

<terminated> TestArray [Java Application] /Library/Java/JavaVirtualMachines/jdk-12.0.2.jdk
How many elements in the array? 10
The biggest element in the array? 100
Before sorting: [93, 31, 62, 56, 61, 85, 99, 13, 18, 4]
After sorting: [4, 13, 18, 31, 56, 61, 62, 85, 93, 99]

```

2、第四章习题 9:

创建一个日期类 MyDate，要求可以设置日期、增加日期以及输出日期。

(1) 代码

```

1. package pkg;
2.
3. import java.util.*;
4.
5. public class MyDate {
6.
7.     public static void main(String[] args) {
8.         Scanner sc = new Scanner(System.in);
9.         System.out.println("请输入要设置的年月日");
10.        System.out.print("年: ");
11.        int year = sc.nextInt();
12.        System.out.print("月: ");
13.        int month = sc.nextInt();
14.        System.out.print("日: ");
15.        int day = sc.nextInt();
16.        Date dt = new Date();
17.        dt.setDate(year, month, day);
18.        dt.showDate();
19.        dt.addDate();
20.    }
21.
22. }
23.

```

```

24. class Date {
25.     private int year;
26.     private int month;
27.     private int day;
28.     public void setDate(int y, int m, int d) {
29.         year = y;
30.         month = m;
31.         day = d;
32.     }
33.
34.     void showDate() {
35.         System.out.println("今天是" + year + "年
    " + month + "月" + day + "日");
36.     }
37.
38.     void addDate() {
39.         if ((month == 1 || month == 3 || month == 5
    || month == 7 || month == 8 || month == 10) && day =
    = 31) {
40.             month += 1;
41.             day = 1;
42.         } else if ((month == 4 || month == 6 || mont
    h == 9 || month == 11) && day == 30) {
43.             month += 1;
44.             day = 1;
45.         } else if ((month == 2) && (year % 4 == 0 &&
    year % 100 != 0 || year % 100 == 0) && day == 29) {
46.             month += 1;
47.             day = 1;
48.         } else if (month == 12 && day == 31) {
49.             year += 1;
50.             day = 1;
51.         } else {
52.             day++;
53.         }
54.         showDate();
55.     }
56.
57. }

```

(2) 运行截图

```

Console
<terminated> MyDate [Java Application] /Library/Java/JavaVirtualMachines/jdk-12.0.2.jdk/C
请输入要设置的年月日
年: 2018
月: 11
日: 30
今天是2018年11月30日
今天是2018年12月1日

```

3、第四章习题 10:

编写一个类 ArraySort, 该类有一个 int 类型的一位数组 sim 的成

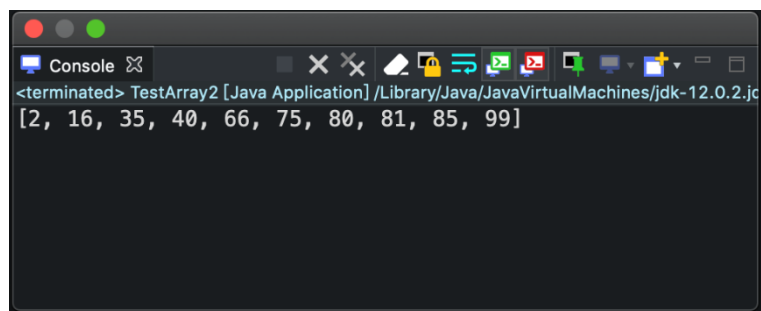
员变量，和一个 `setOrder()` 的成员方法，并有一个带有一个参数的构造方法，该参数将在构造方法中为成员变量数组 `sim` 作初始化。方法 `setOrder()` 不带参数也不返回值，它将成员变量数组 `sim` 的元素从小到大排序。

另有一个类为 `TestArray`，它只有一个 `main` 方法，在该方法中，创建 `ArraySort` 类的对象，并用一个 `int` 类型的一维数组来实例化这个对象的成员变量数组 `sim`。试调用这个对象的 `setOrder()` 方法，并输出其值。

(1) 代码

```
1. package pkg;
2.
3. import java.util.*;
4.
5. class ArraySort {
6.     public int[] sim;
7.     void setOrder() {
8.         Arrays.sort(sim);
9.         System.out.println(Arrays.toString(sim));
10.    }
11.    public ArraySort(int n) {
12.        sim=new int[n];
13.        for (int i = 0; i < n; i++) {
14.            int e = (int) (Math.random() * 100);
15.            sim[i] = e;
16.        }
17.    }
18. }
19.
20. public class TestArray2 {
21.     public static void main(String[] args) {
22.         ArraySort array =new ArraySort(10);
23.         array.setOrder();
24.     }
25. }
```

(2) 运行截图



4、第四章习题 11:

创建一个类 `Point`，有成员变量 `x`, `y`，它们都是 `int` 类型。该类有

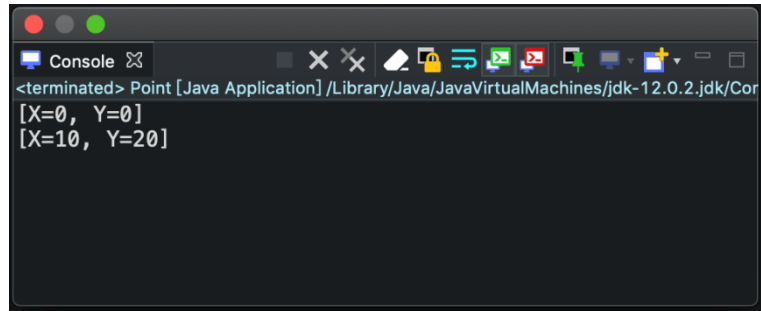
四个成员方法 `setX(int)`、`setY(int)`、`getPoint()`和 `movePoint(int, int)`。`setX(int)`和 `setY(int)`方法是设置成员变量 `x` 和 `y` 的值，`getPoint()`方法则获得由 `x`, `y` 构成的坐标点，`movePoint(int, int)`带两个 `int` 参数，用来修改 `x`, `y` 构成的坐标点。`Point` 类有一个构造方法，不带参数，为 `x`, `y` 设置原点值。

另一个类为 `TestPoint`，有一个 `main()`方法，用来对 `Point` 类的实例进行测试。要求为其实例设置(0, 0)坐标点，再移动到(10, 20)坐标点上，并输出实例调用相应的方法的结果。

(1) 代码

```
1. package pkg;
2.
3. public class Point {
4.     private int x;
5.     private int y;
6.
7.     public void setX(int x) {
8.         this.x = x;
9.     }
10.
11.    public void setY(int y) {
12.        this.y = y;
13.    }
14.
15.    public Point() { // constructor
16.        this.x = 0;
17.        this.y = 0;
18.    }
19.
20.    public Point getPoint() {
21.        return this;
22.    }
23.
24.    public void movePoint(int x, int y) {
25.        this.x = x;
26.        this.y = y;
27.    }
28.
29.    public String toString() {
30.        return "[X=" + x + ", Y=" + y + "]";
31.    }
32.
33.    public static void main(String[] args) {
34.        Point point = new Point();
35.        System.out.println(point);
36.        point.movePoint(10, 20);
37.        System.out.println(point);
38.    }
39. }
```

(2) 运行截图



```
<terminated> Point [Java Application] /Library/Java/JavaVirtualMachines/jdk-12.0.2.jdk/Contents/Home/bin/java
[X=0, Y=0]
[X=10, Y=20]
```

5、实现两个类，分别是长方形和三角形求面积。

(1) 代码

```
1. package pkg;
2.
3. public class Area {
4.     public static void main(String[] args) {
5.         Triangle1 tri = new Triangle1(5, 2);
6.         Rectangle rec = new Rectangle(5, 2);
7.         System.out.println("Trian-
8.         gle Area = " + tri.coutArea(tri.get-
9.         Base(), tri.getHeight()));
10.        System.out.println("Rectan-
11.        gle Area = " + rec.coutArea(rec.get-
12.        Base(), rec.getHeight()));
13.    }
14.}
15.
16.class Triangle1 {
17.    public int base;
18.    public int height;
19.    public float area;
20.
21.    public int getBase() {
22.        return this.base;
23.    }
24.
25.    public int getHeight() {
26.        return this.height;
27.    }
28.
29.    public Triangle1(int base, int height) {
30.        this.base = base;
31.        this.height = height;
32.    }
33.
34.    public float coutArea(int base, int height) {
35.        return (float) (base * height * 0.5);
36.    }
37.}
38.
39.class Rectangle {
40.    public int base;
41.    public int height;
42.    public float area;
43.
44.    public int getBase() {
```

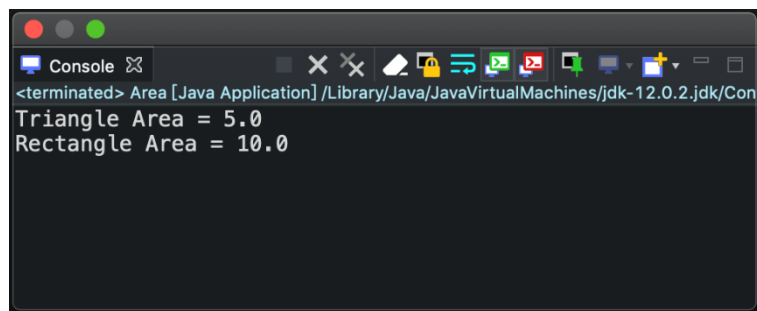


```

41.         return this.base;
42.     }
43.     public int getHeight() {
44.         return this.height;
45.     }
46.     public Rectangle(int base, int height) {
47.         this.base = base;
48.         this.height = height;
49.     }
50.     public float coutArea(int base, int height) {
51.         return (float) (base * height);
52.     }
53. }

```

(2) 运行截图



- 6、编写程序,找出 $1 \sim n$ 以内的所有素数。要求使用数组元素的下标从 $1 \sim n$ 以内表示这些数值,数组元素的值作为素数的标志。其中用数组元素的值为 0 来表示该元素的下标的数值是素数,用 1 来表示该元素的下标的数值不是素数,并输出这些素数。

(1) 代码

```

1. package pkg;
2.
3. import java.util.*;
4. public class IsPrime {
5.     public static boolean isPrimeNorm(int num) {
6.         for (int i = 2; i < num; i++) {
7.             if (num % i == 0) {
8.                 return false;
9.             }
10.        }
11.        return true;
12.    }
13.    public static void main(String[] args) {
14.        long startTime = System.currentTimeMillis();
15.        Scanner in = new Scanner(System.in);
16.        System.out.print("Please input n: ");
17.        int n = in.nextInt();
18.        in.close();
19.        int[] array = new int[n + 1];
20.        for (int i = 0; i < n + 1; i++)
21.            array[i] = 1;

```

```

22.         for (int i = 2; i < n + 1; i++) {
23.             if (isPrimeNorm(i) == true)
24.                 array[i] = 0;
25.         }
26.         int count = 0;
27.         for (int i = 1; i < n + 1; i++) {
28.             if (array[i] == 0) {
29.                 System.out.print(i + "\t");
30.                 count++;
31.             }
32.             if (array[i] == 0 && count % 8 == 0)
33.                 System.out.println();
34.         }
35.         long endTime = System.currentTimeMillis();
36.         System.out.println("\nTotal runtime: " + (endTime - startTime) + "ms");
37.     }
38. }

```

(2) 运行截图

```

<terminated> IsPrime [Java Application] /Library/Java/JavaVirtualMachines/jdk-12.0.2.jdk/Contents/Resources/Java
Please input n: 100
2       3       5       7       11      13       17       19
23      29      31      37      41      43       47       53
59      61      67      71      73      79      83      89
97
Total runtime: 3238ms

```

九、 总结及心得体会：

本实验目的是让学生熟悉 Java 的类、对象的基本操作，理解面向对象程序设计的基本思想。

通过本实验，我体会到了面向对象程序设计的基本思想，掌握了 Java 对数组的排序方法、Java 自带的 Date 类的使用，以及类和对象的基本操作。提高了我对问题的分析能力以及编码能力，受益良多。

十、 对本实验过程及方法、手段的改进建议：

实验内容存在重复的题目，且部分题目的时效性不强。可以采用一些新颖的题目背景，激发学生的编程兴趣、提高实验能力。

报告评分：

指导教师签字：