Java语言程序设计实践环节

考核大纲

**课程名称：Java语言程序设计（实践） 课程代码：02329**

一、考核目标

1.掌握调试、完善和简单设计Java程序的能力；

2.掌握jdk开发工具的使用（新建java源程序，编译与运行程序）；

3.掌握Java的基本语句，基本输入输出流；

4.掌握使用类及方法进行Java面向对象程序开发的方法。

二、考核必备的软硬件条件

1.平台要求：Windows 7，Windows 10等

2.软件要求：jdk , Eclipse

**三、考核要求**

（一）考核内容和要求

1.Java程序的输入、输出语句、基本结构与基础语法；

2.熟悉Java开发环境，实现常见的基本算法。

**（二）考核方式**

采用上机操作考核形式。考核时间为60分钟，采用百分制评分。

**参考示例：（1)字符统计程序**

编写输入字符行，统计输入字符行中数字符、英文字母个数。

Import java.util.Scanner类

public class StaChar

{

 public static void main(String[] args)

 {

 Scanner scan= new Scanner(System.in);

 System.out.println("请输入字符串：");

 String str = scan.nextLine();

 char[] c = str.toCharArray();

 int numberCount = 0;

 int letterCount = 0;

 for (int i = 0; i < c.length; i++) {

 if (c[i] < '9' && c[i] > '0')

 numberCount++;

 else if ((c[i] > 'A' && c[i] < 'Z') || (c[i] > 'a' && c[i] < 'z'))

 letterCount++;

 }

 System.out.println( "输入内容：\n" + str + "\n数字字符：" + numberCount + "个； " + "\n字母：" + letterCount + "个");

 }

}

**参考示例：（2)特殊性质数的判断。**

**素数的判断程序；**

**import** javax.swing.JOptionPane;

/\*\*

 \* 2 找质数程序,利用对话框读入整数,输出2至这个整数之间的质数.<BR>

 \* **@author** 黎明你好

 \*/

**public** **class** PrintPrime

{

 **private** **int** number;// 正整数

 **private** String result = "";

 **public** PrintPrime()// 构造方法

 {

 number = getIntegerNumber("输入整数n", 0);// 要求是>=0的整数

 **if** (number < 0)

 {

 **return**;// 出现错误，程序结束

 }

 **else**

 // 如果大于等于2，开始用循环计算结果

 {

 **for** (**int** i = 2; i <= number; i++) // 计算素数和

 {

 **if** (isPrimeNumber(i))

 result += i + " ";

 }

 }

 // 显示最后的和

 JOptionPane.*showMessageDialog*(**null**, number + "之前所有素数为：\n “" + result + "”", "显示结果",

 JOptionPane.*INFORMATION\_MESSAGE*);

 }

 /\*\*

 \* 通过图形界面，得到符合规则的正整数的方法

 \* **@param** message - 在弹出的对话框中，显示提示信息

 \* **@param** min - 要求此数必须大于等于min

 \* **@return** - 返回符合规则的整数

 \*/

 **public** **int** getIntegerNumber(String message, **int** min)

 {

 String str = JOptionPane.*showInputDialog*(**null**, message, "提示信息",

 JOptionPane.*INFORMATION\_MESSAGE*);

 **int** number = -1;

 **try**

 {

 number = Integer.*parseInt*(str); // 得到输入的正整数

 }

 **catch**( Exception e )

 {

 JOptionPane.*showMessageDialog*(**null**, "输入非数字字符\n程序结束", "错误警告", JOptionPane.*ERROR\_MESSAGE*);

 **return** -1; // 输入的不是数字字符，程序结束

 }

 **if** (number < min)

 {

 JOptionPane.*showMessageDialog*(**null**, "输入的数不符合规则，不是正整数\n程序结束", "错误警告",

 JOptionPane.*ERROR\_MESSAGE*);

 **return** -1; // 输入的数不是大于2的正整数时候，程序结束

 }

 **else**

 **return** number;

 }

 /\*\*

 \* 判断是否是素数的方法

 \* **@param** n - 需要判断的数

 \* **@return** - 是素数返回true，否则返回false

 \*/

 **public** **boolean** isPrimeNumber(**int** n)

 {

 **for** (**int** i = 2; i < n; i++)

 {

 **if** (n % i == 0)

 **return** **false**;

 }

 **return** **true**;

 }

 /\*\* main方法 \*/

 **public** **static** **void** main(String[] args)

 {

 **new** PrintPrime();

 }

}

**参考示例：（**3**)类的继承定义。声明几何形状类，类中定义几何形状的成员变量和方法，然后继承声明几何形状类，创建对象，并显示对象的相关信息。**

/\*\*

 \* 几何图形类，抽象类

 \*/

**abstract** **class** Shape

{

 **public** **abstract void** message();

}

/\*\*

 \* 圆形类

 \*/

**class** Circle **extends** Shape

{

 **private** **float** R;

 **public** Circle(**float** r)

 {

 R = r;

 }

 **public** **void** message()

 {

 **System.out.println(“圆的半径是”+r)**;

 }

}

/\*\*

 \* 矩形类

 \*/

**class** Rectangle **extends** Shape

{

 **private** **float** w, h;

 **public** Rectangle(**float** w, **float** h)

 {

 **this**.w = w;

 **this**.h = h;

 }

 **public** **void** message()

 {

 **System.out.println(“矩形宽是”+w+“矩形高是”+h)**;

 }

}

**public** **class** Work11\_3

{

 **public** **static** **void** main(String args[])

 {

 Circle c;

 Rectangle r;

 c = **new** Circle(2.0f);

 r = **new** Rectangle(3.0f, 5.0f);

 c.message();

 r.message();

 }

}

### **参考示例：（**4**）类的多态练习。**

### **在示例3的基础上，给出每个几何形状类中计算面积的方法。要体现对象多态的概念。**

/\*\*

 \* 几何图形类，抽象类

 \*/

**abstract** **class** Shape

{

 **public** **float** area()

 {

 **return** 0.0f;

 }

}

/\*\*

 \* 圆形类

 \*/

**class** Circle **extends** Shape

{

 **private** **float** R;

 **public** Circle(**float** r)

 {

 R = r;

 }

 **public** **float** area()

 {

 **return** (**float**) (Math.*PI* \* R \* R);

 }

}

/\*\*

 \* 矩形类

 \*/

**class** Rectangle **extends** Shape

{

 **private** **float** w, h;

 **public** Rectangle(**float** w, **float** h)

 {

 **this**.w = w;

 **this**.h = h;

 }

 **public** **float** area()

 {

 **return** w \* h;

 }

}

**public** **class** Work11\_4

{

 **public** **static** **void** main(String args[])

 {

 Circle c;

 Rectangle r;

 c = **new** Circle(2.0f);

 r = **new** Rectangle(3.0f, 5.0f);

 System.*out*.println("圆面积" + *returnArea*(c));

 System.*out*.println("长方形面积" + *returnArea*(r));

 }

 **static** **float** returnArea(Shape s)

 {

 **return** s.area();

 }

}