**多重循环、方法作业**

1. **选择题**

|  |  |  |
| --- | --- | --- |
| **1.** | **分析下面的Java多重循环代码片段，编译运行后的输出结果是（D ）。（选择一项）** | |
|  | **for (int i = 0; i < 6; i++) {**  **int k = ++i; // 1 3 5**  **while (k < 5) {**  **System.*out*.print(i); // 13**  **break;**  **}**  **}** | |
|  |  |  |
|  | **A.** | **024** |
|  | **B.** | **02** |
|  | **C.** | **123** |
|  | **D.** | **13** |

|  |  |  |
| --- | --- | --- |
| **2.** | **给定如下Java代码，编译运行的结果是（ A）。（选择一项）** | |
|  | **public class Test {**  **public static void main(String[] args) {**  **int sum=0;**  **for(int i=1;i<10;i++){**  **do{**  **i++; // 2 3 4 5 6**  **if(i%2!=0)**  **sum+=i; // 3 8**  **}while(i<6);**  **}**  **System.*out*.println(sum);**  **}**  **}** | |
|  |  |  |
|  | **A** | **8** |
|  | **B.** | **15** |
|  | **C.** | **24** |
|  | **D.** | **什么也不输出** |

|  |  |  |
| --- | --- | --- |
| **3.** | **Java中main方法的返回值是（ D ）。（选择一项）** | |
|  |  |  |
|  | **A** | **String** |
|  | **B.** | **int** |
|  | **C.** | **char** |
|  | **D.** | **void** |

|  |  |  |
| --- | --- | --- |
| **4.** | **在Java的程序类中如下方法定义正确的是（ AD ）。（选择两项）** | |
|  |  |  |
|  | **A** | **public int ufTest(int num){**  **int sum=num+100;**  **return sum;**  **}** |
|  | **B.** | **public String ufTest(int num){**  **int sum=num+100;**  **return sum;**  **}** |
|  | **C.** | **public void ufTest(int num){**  **int sum=num+100;**  **return sum;**  **}** |
|  | **D.** | **public float ufTest(int num){**  **int sum=num+100;**  **return sum;**  **}** |

|  |  |  |
| --- | --- | --- |
| **5.** | **以下关于方法调用的代码的执行结果是（ B ）。（选择一项）** | |
|  | **public class Test {**  **public static void main(String args[]) {**  **int i = 99;**  ***mb\_operate*(i);**  **System.*out*.print(i + 100);**  **}**  **static void mb\_operate(int i) {**  **i += 100;**  **}**  **}** | |
|  |  |  |
|  | **A.** | **99** |
|  | **B.** | **199** |
|  | **C.** | **299** |
|  | **D.** | **99100** |

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| **6.** | **下面（ A D ）方法是public void example(){…}的重载方法。（选择二项）** | |
|  |  |  |
|  | **A** | **private String example(){…}** |
|  | **B.** | **public int example(String str){…}** |
|  | **C.** | **public void example2(){…}** |
|  | **D.** | **public int example(int m,float f){…}** |

|  |  |  |
| --- | --- | --- |
| **7.** | **以下选项中添加到代码中横线处会出现错误的是（ B D ）。（选择二项）** | |
|  | **public class Test {**  **public float aMethod(float a, float b) {**  **return 0;**  **}**    **}** | |
|  |  |  |
|  | **A** | **public float aMethod(float a, float b, float c) {**  **return 0;**  **}** |
|  | **B.** | **public float aMethod(float c, float d) {**  **return 0;**  **}** |
|  | **C.** | **public int aMethod(int a, int b) {**  **return 0;**  **}** |
|  | **D.** | **private int aMethod(float a, float b) {**  **return 0;**  **}** |

1. **判断题**
2. **多重循环是指一个循环体内又包含另一个完整的循环结构。外层循环变量变化一次，内层循环变量要从头到尾变化一遍。（ T ）**
3. **定义方法时的参数称为实在参数，调用方法时的参数称为形式参数。（F ）**
4. **调用方法时，要求实参和形参的个数相同，类型匹配。（ T ）**
5. **判断方法重载的依据是方法名相同，参数不同。（ F ）**
6. **简答题**
   1. **在多重循环中，如何在内层循环中使用break跳出外层循环。**

**加一个outer**

outer:

**for**(**int** i=1;i<=10;i++) {

**for**(**int** j=1;j<=5;j++) {

**if**(j==2) {

**System**.***out***.print(j);

**break** outer;

}

}

**System**.***out***.println();

}

* 1. **方法重载的定义、作用和判断依据**

**方法的重载：在同一个类中，方法名相同，参数列表不同**

**作用：同一个方法名，不同的作用**

**判断依据：方法名相同，返回类型相同，参数列表不同（形参的类型，个数，顺序不同）**

* 1. **方法的定义中，是否必须要有返回值类型?如果确实不需要返回值，使用哪个关键字声明?**

**构造方法不需要返回值，void**

* 1. **方法的定义中，return是否必须?它有什么作用?**

**不是，返回值类型**

* 1. **java中，参数的传递使用值传递还是引用传递?**

**引用**

* 1. **方法定义中，形式参数和实际参数有什么区别?用自己的话描述。**

**形式参数：是方法中传递的参数**

**实际参数：是主函数传递的函数**

* 1. **方法定义时，使用实参还是形参?**

**形参**

* 1. **定义形参时，必须要写变量类型吗?**

**是**

* 1. **实参类型是否必须和形参类型匹配?**

**是**

* 1. **模仿老师讲“网吧上网的例子”，讲一个自己的事例来说明方法的定义，形参和实参。**

1. **编码题**
   1. **打印九九乘法表：**

**package** homework;

**public** **class** **Test01** {

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

//打印九九乘法表

**for** (**int** i = 1; i <=9; i++) {

**for** (**int** j = 1; j <=i; j++) {

**System**.***out***.print(j+"\*"+i+"="+i\*j+"\t");

}

**System**.***out***.println();

}

}

}

* 1. **判断1-100之间有多少个素数并输出所有素数。（将判断一个数是否是素数的功能提取成方法，在循环中直接调用即可）**

**package** homework;

**public** **class** **Test02** {

**public** **static** **boolean** **prime**(**int** num) {

**boolean** flag=**true**;

**for**(**int** j=2;j<num;j++)

**if**(num%j==0) {

flag=**false**;

**break**;

}

**return** flag;

}

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

//判断1-100之间有多少个素数并输出所有素数。（将判断一个数是否是素数的功能提取成方法，在循环中直接调用即可）

**int** count=0;

**for**(**int** i=1;i<=100;i++) {

**boolean** flag=*prime*(i);

**if**(flag) {

count++;

**System**.***out***.println(i);

}

}

**System**.***out***.println("素数有："+count);

}

}

* 1. **输入三个班，每班3个学生的成绩，求和并求平均分**

**package** homework;

**import** java.util.Scanner;

**public** **class** **Test03** {

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

//输入三个班，每班3个学生的成绩，求和并求平均分

**Scanner** sc=**new** Scanner(**System**.***in***);

//sum总和,count计数器

**int** sum=0,count=0;

**int**[][] score=**new** **int**[3][3];

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

//sum1班级和

**int** sum1=0;

**for** (**int** j = 0; j < score[i].length; j++) {

**System**.***out***.print((i+1)+"班的第"+(j+1)+"名同学的成绩：");

score[i][j]=sc.nextInt();

sum1+=score[i][j];

count++;

}

sum+=sum1;

**System**.***out***.println((i+1)+"班的成绩："+sum1);

**System**.***out***.println((i+1)+"班的平均分："+sum1/score[i].length);

}

**System**.***out***.println("3个班的总成绩："+sum);

**System**.***out***.println("3个班的平均分："+sum/count);

}

}

1. **可选题**
   1. **打印出所有的"水仙花数"，所谓"水仙花数"是指一个三位数，其各位数字立方和等于该数本身。例如：153是一个"水仙花数"，因为153=1的三次方＋5的三次方＋3的三次方。**

**package** homework;

**public** **class** **Test04** {

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

//打印出所有的"水仙花数"，所谓"水仙花数"是指一个三位数，其各位数字立方和等于该数本身。

//例如：153是一个"水仙花数"，因为153=1的三次方＋5的三次方＋3的三次方。

**for**(**int** i=100;i<=999;i++) {

**if**(i==(i/100)\*(i/100)\*(i/100)+(i/10%10)\*(i/10%10)\*(i/10%10)+(i%10)\*(i%10)\*(i%10))

**System**.***out***.print(i+"\t");

}

}

}

* 1. **定义方法：打印指定行和列的矩形**

**package** homework;

**import** java.util.Scanner;

**public** **class** **Test05** {

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

**Scanner** sc=**new** Scanner(**System**.***in***);

**System**.***out***.print("请输入矩阵的行：");

**int** hang=sc.nextInt();

**System**.***out***.print("请输入矩阵的列：");

**int** lie=sc.nextInt();

**int**[][] arr=**new** **int**[hang][lie];

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

**for** (**int** j = 0; j < arr[i].length; j++) {

**System**.***out***.print("\*");

}

**System**.***out***.println();

}

}

}