一、填空题

1.函数重载是指在函数名相同，但\_接受的形参的个数或类型\_不同

2.创建大小为2行4列的二维char型数组的语句为\_\_ char[][] s = new char[2][4];\_\_，数组创建后每个元素的值为\_\_’\u0000’\_\_。

3. 创建一个大小为10的整型数组，且数组元素的值分别为1,2,3,4,5,6,7,8,9,10的语句为\_\_ int[] num = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };\_\_\_\_\_\_。

4. 用final关键字修饰一个方法形参的含义是\_\_\_\_形参在方法中不会被修改\_\_\_\_\_\_\_。

5. 下列程序存在的错误是\_\_\_\_\_\_参数重复 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。public static void m(int i){

for(int i = 0 ; i < 10; i++){

System.out.println(i);

}

}

1. 单项选择题

1.下列语句的输出结果是\_c\_\_。

String[][] a = {

{"Beijing","Wuhan"},

{"Shanghai","Guangzhou","Xian"},

{"Chongqing","Chengdu"}

};

System.out.println(a[a.length - 1].length);

System.out.println(a[a.length - 1][a[a.length - 1].length - 1].length());

A．2, 5 B．3, 4 C．2, 7 D．3, 8

2. String[]s={“Monday”,”Tuesday”,”Wednesday”,”Thirsday”,”Friday”,”Sataday”,”Sunday”}；，则下列语句正确的是\_\_\_c\_。

A．int a = s.length, b = s[1].length;

B．int a = s.length( ), b = s[1].length;

C．int a = s.length, b = s[1].length();

D．int a = s.length( ), b = s[1].length( );

3. 若有下面程序

class C {

public static void main(String[] args) {

int[] array = new int[10];

increase(array);

System.out.print(array[0]);

}

public static void increase(int[] array) {

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

array[i]++;

}

}

}

则输出为\_\_\_b\_\_\_。

A.0 B.1 C.2 D.10

4. 下面的数组申明和初始化语句不合法的是\_\_c\_\_\_。

A. int a[ ] = null；

B. int[ ] b = { };

C. int[ ] c = new int{1,2,3,4};

D. int [] d[] = new int[5][ ];

三、判断对错题

1. 局部变量在使用前必须通过初始化或者赋值语句显式地给一个值。(√)

2. 一个方法必须要有一个return语句。(×)

3. 如果定义int[] nValues={1,2,3,4}; 那么nValues为引用类型。(√)

4. 不能基于函数返回类型来重载函数。（√）

5. 二维数组的行数和列数是相同的。（×）

四、阅读下列程序，写出输出结果：

public class Test2 {

public static void main(String[] args){

int[] a = {1};

String[] s = {"Hello"};

int i = a[0];

m(s,a,i);

for(String v:s){

System.out.println(v);

}

for(int v:a){

System.out.println(v);

}

System.out.println(i);

}

public static void m(String[] a1, int[] a2, int i){

for(int j =0; j < a1.length;j++){

a1[j] = "Java";

}

for(int j =0; j < a2.length;j++){

a2[j]++;

}

i++;

}

}

Answer：

Java

2

1

五、编程题

1：实现下面二个方法，并在Test3里添加入口main函数测试运行。

Tips：注意检查输入参数row的值，当输入负数，0时如何处理也考虑进来，如何处理这种情况不做要求，可以简单地打印出提示信息，或者抛出异常。但最简单的办法就是当出现这些边界条件，直接返回null引用就行了。由这个方法的调用者去处理。另外也不考虑当row的值太大导致内存溢出的情况。

public class test3 {

    public static  int[][] createArray(int row){

        if(row<=0){

            return null;

        }

        int x[][]=new int[row][];

        for(int i=0;i<row;i++){

            x[i]=new int [row-i];

        }

        return x;

}

    public static  void printArray(int[][] a){

        if(a==null){

            System.out.print("Peremeter error");

            return;

        }

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

            for(int k=0;k<a[i].length;k++){

                System.out.print(a[i][k]+" ");

            }

            System.out.print("\n");

        }

    }

    public static void main(String[] args){

        System.out.println("Enter the column num of the matrix you want to create:");

        Scanner scanner = new Scanner(System.in);

        int tmp=scanner.nextInt();

        printArray(createArray(tmp));

        scanner.close();

    }

}

测试结果（包含0，正数和负数）：

