

东北大学考试试卷 (A 卷)

2003—2004 学年第 学期

课程名称: Java 程序设计

总分	一	二	三	四	五	六	七	八	九	十		

院
级
号
名

得分
----

密  
封  
线

一. 填空

1. (6分) 最显著的优点: (平台无关 可移植); 内存的自行管理: (垃圾回收);  
(简洁性) 取消了头文件、指针算法、结构、单元、运算符重载、虚基类等概念;

(单编译器) 编译器在中间过程生成字节码指令, 可以易于任何机器上的解释;

Java 虚拟机: (指令集), (寄存器组), (类文件的流 (栈结构)), (垃圾回收) 存储区六部分;

( // ) 表示注释以行; ( /\* \*/ ) 表示注释一段;

2. (1分) 下面变量定义那个错误? ( C )  
(A) byte b1; (B) int tem2p; (C) long 2head; (D) char wm;

3. (3分) 给出结果 15/4=( 3 ), 15%2=( 1 ), 11.0/4=( 2.75 );

4. (4分) 求程序中的 a, b, c, d 运行后的结果?

```
int a = 10;
int b = 10;
int d = 0;
double c = 6.36;
a = a * (b++);
d = (int) (-b * c);
```

a = 100    b = 11  
b = 10    d = 63

5. (2分) 求程序中的 n, message 运行后的结果?

```
String exp = "Expletive ";
String pg = "deleted";
String message = exp + pg;
```

```
String greeting="Hello";
int n=greeting.length();
```

5

6. (1分) 求程序中的 n 运行后的结果?

```
Public static void main(String[] args)
{
    int n=0;
    [ int k;
    int n=1; //?
    ]
}
```

编辑不过  
局部变量定义

7. (3分) 求下面三个程序中的 i 运行后的结果?

```
int i=0;
while(i!=0) {i++;}
```

```
int i=0;
do {i++;} while(i!=0);
```

```
int i=0;
for(int i=1; i<=10; i++);
```

0  
编译不过 / 无限死循环, 有死溢  
局部变量要定义.

8. (2分) 求下面程序中的 i, j 运行后的结果?

```
public class BreakAndContinue {
    public static void main(String[] args) {
        for(int i = 0; i < 100; i++) {
            if(i == 74) break;
            if(i % 9 != 0) continue;
        }
        int i = 0;
        while(true) {
            i++;
            int j = i * 27;
            if(j == 1269) break;
            if(i % 10 != 0) continue;
        }
    }
}
```

i = 74    j = 1269  
27 / 1269  
108  
189  
139

9. (2分) ( 封装 ): 对每个对象的处理需求, 和属性一起封装并被视为一个固有的整体。

( 接口 ): 对象之间一个较窄的、定义明确的界面。  
( 封装 ) 属性及其服务的封装, 通过问题空间的若干实例对问题空间抽象。  
面向对象=对象+分类+( 继承 )+通过消息的通信;

10. (5分) 求下面程序中的 count1~5 运行后的结果?

```
public class Robot {
```

```

public Robot(int x, int y) {
    this.x = x;
    this.y = y;
}
public void move(int dx, int dy) {
    x += dx; y += dy;
    instanceMoveCount++;
    classMoveCount++;
}
public int getInstanceMoveCount() {
    return instanceMoveCount;
}
public static int getClassMoveCount() {
    return classMoveCount;
}
public int x, y;
private int instanceMoveCount;
static private int classMoveCount;
public static void main(String[] args) {
    Robot r1 = new Robot(0, 0);  X=0, y=0
    Robot r2 = new Robot(0, 0);  X=0, y=0
    r1.move(20, 10);  X=20, y=10,  instance=1, class=1
    r1.move(10, 20);  X=30, y=30,  instance=2, class=2
    r2.move(10, 10);  X=10, y=10,  instance=1, class=3
    int count1 = r1.getInstanceMoveCount();  2
    int count2 = r2.getInstanceMoveCount();  1
    int count3 = r1.getClassMoveCount();  3
    int count4 = r2.getClassMoveCount();  3
    int count5 = Robot.getClassMoveCount();  3
}
}
    
```

11. (2分) 给出下面程序的输出结果:

```

public class Minimum {
    int Mine(int i, int j)
    {
        if (i>j) return j;
        else return i;
    };
    public static void main(String[] args) {
        int a[] = { 75, 34, 80, 11, 95, 34, 53, 81, 33, 13 };
        int min = a[0];
        for (int i = 1; i < a.length; i++) {
            min = Mine(min, a[i]);
        }
    }
}
    
```

min=11

```

]
System.out.println("The minimum value is: " + min);
}
    
```

The minimum value is 11

12. (2分) 输出下面程序的结果:

```

public class OverloadingOrder {
    static void print(String s, int i) {
        System.out.println(
            "String: " + s +
            ", int: " + i);
    }
    static void print(int i, String s) {
        System.out.println(
            "int: " + i +
            ", String: " + s);
    }
    public static void main(String[] args) {
        print("String first", 11);
        print(99, "Int first");
    }
}
    
```

String: string first, int: 11  
int: 99, string: int first

13. (3分) 输出下面程序的结果:

```

public class Parcel6 {
    public Contents cont() {
        return new Contents() {
            private int i = 11;
            public int value() { return i; }
        }; // Semicolon required in this case
    }
    public static void main(String[] args) {
        Parcel6 p = new Parcel6();
        Contents c = p.cont();
    }
}
    
```

内部类

14. (5分) 输出下面程序的结果:

```

class TT{
    {
        System.out.println("TT");
    }
}

public class RandomIntGenerator {
    public RandomIntGenerator(int l, int h){
        low=l;
        high=h;
        System.out.println("#|");
    }
    public int draw(){
        int r=low+(int)((high-low+1)*nextRandom());
        return r;
    }
    public static void main(String[] args){
        TT t1=new TT();
        TT t2=new TT();
        RandomIntGenerator r1=new RandomIntGenerator(1,10);
        RandomIntGenerator r2=new RandomIntGenerator(0,1);
    }
    private static double nextRandom(){
        int pos=(int)(java.lang.Math.random()*BUFFER_SIZE);
        if(pos==BUFFER_SIZE) pos=BUFFER_SIZE-1;
        double r=buffer[pos];
        buffer[pos]=java.lang.Math.random();
        return r;
    }
    private static final int BUFFER_SIZE=11;
    private static double[] buffer=new double[BUFFER_SIZE];
    static
    {
        int i;
        for(i=0; i<BUFFER_SIZE; i++)
            buffer[i]=java.lang.Math.random();
        System.out.print("*|");
    }
    private int low;
    private int high;
}
    
```

得分

二. (1分) 使用封装的类有两种方法, 假设有 Java.www.make 对象, 请用任何一种方法为这个对象定义一个实例对象。

```
Java.www.make m = new Java.www.make();
```

得分

三. (5分) 修改上例, 使 A 和 B 都具有参数的构造函数, 为 C 编写一个构造函数 (输入参数为空), 并且为 C 创建一个实例。

考:

得分

四. (3分) 定义一个“鸟”的类: 简单描述鸟的属性特征 (羽毛, 嘴, 爪), 和它的基本行为 (飞, 跳)

```

public class Bird {
    private String feather;
    private String mouth;
    private String claw;
    public void fly() { }
    public void jump() { }
}
    
```

mouth  
feather claw  
fly jump

得分

五. (5分) 编写一个程序, 编译完成后, 做如下:  
java 程序名 姓名  
程序执行后, 返回“姓名”?

```

public class Test {
    public static void main(String[] args) {
        System.out.println(args[0]);
    }
}
    
```

得分

六. (5分) 将一个集合 {1, 16, 5, 7, 20, 37, 36, 66} 有从小到大进行排列?

```

public class Sort {
    public static void main(String[] args) {
        int[] sortBefore = {1, 16, 5, 7, 20, 37, 36, 66};
    }
}
    
```

得分

七. (5分) 请编写两个带有构造函数(输入参数为空)的类A和B, 并且产生类C, 使其继承A, 并在C中以B为成员, 不要为C提供任何构造函数, 最后, 产生一个C的实例观察结果。

```

class A {
    A(int i) {}
}

class B {
    public void static main() {}
}

public class C {
    class B b;
}
    
```

是否可以使用A("James", 100, 3)来初始化类A中的三个变量(name, hirDay, salary), 如果不能如何改正?

不可以  
 ① 更改构造函数  
 ② 强制转化

得分

八. 改错

1. (1分) 变量初始化的位置正确么?

```

int test(int testval, int target) {
    if(testval > target)
        result = +1;
    else if(testval < target)
        result = -1;
    else
        result = 0; // Match
    return result;
    int result = 0;
}
    
```

代码不可达



2. (1分) 现有两个类A和B, 其中A类继承了B类, 即 class A extends B {}; 问下面的表达示是否正确?

```

A a=new A();
B b=a;
    
```

正确

3. (3分)

```

class A {
    String name;
    int hirDay;
    double salary;

    A(String n, double s, int d)
    {
        name = n;
        hireDay = d;
        double salary = s;
    }
}
    
```

不可以

name = n  
 hirDay = d  
 salary = s