

## 一、单选题

1	2	3	4	5	6	7	8	9	10
C	C	D	D	A	A	B	B	A	D
11	12	13	14	15	16	17	18	19	20
C	D	B	A	C	C	B	C	A	A
21	22	23	24	25	26	27	28	29	30
D	D	D	D	B	C	D	C	C	B

## 二、填空题

1. JavaSE    JavaE    2. 字节码    .class    3. 封装    继承  
 4. 抽象 (或 abstract)    final    5. super    6. 接口

## 三、阅读理解题

1. 6    3. 10.0  
 2. 1    4. a=60  
      4    b=20  
      9    c=30  
 Tottle is 14    5. 234

## 四、编程题

1. 【1】 extends Point    【2】 super(a1,b1)    【3】 Rec r1=new Rec(0,0,10,20)

```
2. import java.util.Arrays;
public class CharsSort {
    public static void main(String[] args) {
        char[] chars = new char[]{'a','c','u','b','e','p','f','z'};
        System.out.print("原字符序列: ");
        for(int i = 0; i < chars.length; i++){
            System.out.print(chars[i] + " ");
        }
        Arrays.sort(chars);           //对数组进行升序排序
        System.out.print("\n 升序排序后: ");
        for(int i = 0; i < chars.length; i++){
            System.out.print(chars[i] + " ");
        }
        System.out.print("\n 逆序输出为: ");
        for(int i = chars.length-1; i >= 0; i--){
            System.out.print(chars[i] + " ");
        }
    }
}
```

```
3. public abstract class MotoVehicle {
    private String no;// 汽车牌号
    private String brand;// 汽车品牌
    public MotoVehicle() {
    }
    public MotoVehicle(String no, String brand) {
```

```

        this.no = no;
        this.brand = brand;
    }
    public String getNo() {
        return no;
    }
    public String getBrand() {
        return brand;
    }
    public abstract int calRent(int days);
}

public final class Car extends MotoVehicle {
    private String type;// 汽车型号
    public Car() {
    }
    public Car(String no, String brand, String type) {
        super(no, brand);
        this.type = type;
    }
    public String getType() {
        return type;
    }
    public void setType(String type) {
        this.type = type;
    }
    public int calRent(int days) {
        if ("1".equals(type)) { // 代表 550i
            return days * 500;
        } else if ("2".equals(type)) { // 2 代表商务舱 GL8
            return 600 * days;
        } else {
            return 300 * days;
        }
    }
}

public final class Bus extends MotoVehicle {
    private int seatCount;// 座位数
    public Bus() {
    }
    public Bus(String no, String brand, int seatCount) {
        super(no, brand);
        this.seatCount = seatCount;
    }
    public int getSeatCount() {
        return seatCount;
    }
    public void setSeatCount(int seatCount) {
        this.seatCount = seatCount;
    }
    public int calRent(int days) {
        if (seatCount <= 16) {
            return days * 800;
        } else {

```

```

        return days * 1500;
    }
}
import java.util.Scanner;
public class TestRent {
    public static void main(String[] args) {
        String no,brand,mtype,type;
        int seatCount,days,rent;
        Car car;
        Bus bus;
        Scanner input = new Scanner(System.in);
        System.out.println("欢迎您来到汽车租赁公司！");
        System.out.print("请输入要租赁的天数：");
        days=input.nextInt();
        System.out.print("请输入要租赁的汽车类型（1：轿车      2、客车）：");
        mtype = input.next();
        if("1".equals(mtype)){
            System.out.print("请输入要租赁的汽车品牌（1、宝马      2、别克）：");
            brand=input.next();
            System.out.print("请输入轿车的型号 ");
            if("1".equals(brand))
                System.out.print("（1、550i）：");
            else
                System.out.print("（2、商务舱 GL8      3、林荫大道）");
            type=input.next();
            no="京 BK5543";//简单起见，直接指定汽车牌号
            System.out.println("分配给您的汽车牌号是："+no);
            car =new Car(no,brand,type);
            rent=car.calRent(days);
        }
        else{
            System.out.print("请输入要租赁的客车品牌（1、金杯 2、金龙）：");
            brand=input.next();
            System.out.print("请输入客车的座位数：");
            seatCount=input.nextInt();
            no="京 AU8769";//简单起见，直接指定汽车牌号
            System.out.println("分配给您的汽车牌号是："+no);
            bus=new Bus(no,brand,seatCount);
            rent=bus.calRent(days);
        }
        System.out.println("\n 顾客您好！您需要支付的租赁费用是"+rent+"。");
    }
}

```



## 一、单选题

1. 在下列概念中, Java 语言只保留了 D

- A. 运算符重载      B. 结构和联合      C. 指针      D. 方法重载

2. 下列类定义中, 不正确的是 B

- A. `class x { .... }`      B. `static class x implements y1,y2 { .... }`  
C. `public class x extends Applet { .... }`      D. `class x extends y { .... }`

3. 下列语句序列执行后, k 的值是 B

```
int m=3, n=6, k=0;

while( (m++) < (--n) ) ++k;
```

先把 m 的初值赋给表达式 (m++) 作为表达式 (--n) 的值, 然后 m 自增 1 作为 n 的值。

如 i=1, 则表达式 (i++) 的值等于 1, 而 i 的值变为 2。

先自增 1, 再把自增后的值赋给表达式 (i++)。

如 i=1, 则表达式 (i++) 的值等于 2, i 的值也为 2。

- A. 0      B. 1      C. 2      D. 3

4. 设 i、j 为 int 型变量名, a 为 int 型数组名, 以下选项中, 正确的赋值语句是 A

- A. `a[0] = 7;`      B. `i = i + 2`      C. `i++ --j;`      D. `a(0) = 66;`

5. Java 语言的类间的继承关系是 B

- A. 多重的      B. 单重的      C. 线程的      D. 不能继承

6. 下列语句序列执行后, k 的值是 C

```
int i=10, j=18, k=30;

switch( j - i ) {

    case 8: k++;

    case 9: k+=2;

    case 10: k+=3;

    default: k/=j;

}
```

- A. 31      B. 3      C. 2      D. 33

7. 设有定义 `float x=3.5f, y=4.6f, z=5.7f;` 则以下的表达式中, 值为 true 的是 C

- A. `x > y || x > z`      B. `z > (y + x)`      C. `x != y`      D. `x < y & ! (x < z)`

8. 下列修饰符中, 定义类头时能使用的修饰符是 C
- A. private      B. static      C. abstract      D. protected
9. 设 i、j、k 为类 x 中定义的 int 型变量名, 下列类 x 的构造函数中不正确的是 B
- A. x(int m){ ... }      B. void x(int m){ ... }
- C. x(int m, int n){ ... }      D. x(int h, int m, int n){ ... }
10. 下列对封装性的描述中, 错误的是 D
- A. 封装体包含了属性和行为
- B. 封装使得抽象的数据类型提高了可重用性
- C. 被封装的某些信息在封装体外是不可见的
- D. 封装体中的属性和行为的访问权限是相同的
11. 下列关于包的描述中, 错误的是 B
- A. 包是若干个类的集合      B. 包是一种特殊的类型
- C. 包是使用 package 语句创建的      D. 包有有名包和无名包两种
12. 下列关于 Applet 程序的描述中, 错误的是 C
- A. Applet 程序的主类必须是 Applet 类得子类
- B. Applet 不是完整的独立程序
- C. Applet 程序的主类中应有一个 main() 方法
- D. Applet 的字节码文件必须嵌套在一个 HTML 文件中
13. 下列界面元素中, 不是容器的是 C
- A. Dialog      B. Frame      C. List      D. Panel
14. 下列关于异常处理的描述中, 错误的是 D
- A. 程序运行时异常由 Java 虚拟机自动进行处理
- B. 使用 try-catch-finally 语句捕获异常
- C. 使用 throw 语句抛出异常
- D. 捕获到的异常只能在当前方法中处理, 不能在其他方法中处理
15. 以下方法不属于迭代器 (Iterator) 接口所定义的方法是 D
- A. hasNext()      B. next()      C. remove()      D. nextElement()
16. 主线程属于 (C)
- A. 守护线程      B. Thread 线程      C. 用户线程      D. Runnable 线程

## 二、填空题

1. 三目运算符的3个操作数中,第一个操作数的类型必须是 逻辑 型的。
2. 在Java的基本数据类型中,char型采用Unicode编码方案,每个Unicode码占用 2 字节内存空间,这样,无论是中文字符还是英文字符,都是占用 2 字节内存空间。
3. 假设  $x=13$ ,  $y=4$ , 则表达式  $x\%y \neq 0$  的值是 true, 其数据类型是 布尔型。
4. Java语言中,多态性主要表现在两个方面: 方法重载 和 成员覆盖。
5. 定义类的关键字是 class, 定义接口的关键字是 interface。
6. 在Java语言中,通过类的定义只能实现 单 重继承,但通过接口的定义可以实现 多 重继承关系。
7. 在Java中字符串包括字符串常量和字符串变量,分别由类 String 和类 StringBuffer 来创建对象。
8. Java源程序文件编译后产生的文件称为 .class 文件,其扩展名为 .java。
9. 在多线程程序设计中,若要启动一个线程需要调用的函数是 start。
10. 在Java语言中,系统规定:表示当前类的构造方法用 this,表示直接父类的构造方法用 super。
17. Java语言规定标识符是由 字母、下划线、数字 和 美元符号 组成的字符序列,长度不限;又规定第一个字符不能是 数字,并区分字母的大小写。
18. Java中实现多线程一般使用两种方法,一是 Thread,二是 Runnable。
19. Java中变量有三要素: 类型、变量名 和 变量值。
20. Java语言中,说明或声明数组时 不 内存大小,创建数组时 指定 内存大小。
21. 集合框架提供的Map接口的实现类中,在Map中插入、删除和定位元素,使用 HashMap 是最好的选择;需要按顺序遍历键,需选择 TreeMap。
22. OutputStream 类和 InputStream 类是所有字节流输入输出类的父类。
23. 导入 my.package 包中的所类的命令是 import。
24. 构造方法是一种特殊的成员方法,构造方法名与 类 相同,并且此方法 无 有返回值。
25. 图形用户界面的设计中有事件处理模型处理事件,在该模型中主要包括 事件源、事件对象 和 事件监听者。

## 三、判断题

1. Java语言具有较好的安全性和可移植性及与平台无关等特性。T

2. Java 语言的基本数据类型有 4 种：整型、浮点型、字符型和布尔型。T
3. 组成 Java Application 的若干类中，有且仅有一个主类，只有主类中含有主方法 main。  
T
4. 数据由高类型转换到低类型时候，采用强制转换，数据精度要受到损失。T
5. 子类所继承父类的成员都可以在子类中访问。
6. 静态方法只能处理静态变量。T
7. 抽象方法是一种只有说明而无具体实现的方法。
8. Java 语言规定在嵌套的程序块中不允许定义同名的成员变量。
9. Throwable 类有两个子类：Error 类和 Exception 类。前者由系统保留，后者供应用程序使用。
10. 运行 Applet 程序是使用 AppletViewer 命令运行嵌入了字节码文件的 html 文件，从而获得运行结果。
11. 成员方法的重载和覆盖是同一回事。
12. 集合 Set 是通过键-值对的方式来存储对象的。
13. Java 语言是一种强类型语言，数据类型转换有两种：隐含转换和强制转换。
14. Java 语言中，数组在静态和动态赋值时都判越界。
15. Frame 容器是有边框的容器，它也是一种独立窗口，只能作为最外层容器。
16. 死锁的产生是因为多个线程间存在资源竞争。

1. 组成 Java Application 的若干类中，有且仅有一个主类，只有主类中含有主方法 main。( )
2. 数据由高类型转换到低类型时候，采用强制转换，数据精度要受到损失。( )
3. 成员方法的重载和覆盖是同一回事。( )
4. 抽象方法是一种只有说明而无具体实现的方法。( )
5. Java2 的事件处理方法是采用委托事件处理模型。( )
6. Throwable 类有两个子类：Error 类和 Exception 类。前者由系统保留，后者供应用程序使用。  
( )
7. 运行 Applet 程序是使用 AppletViewer 命令运行嵌入了字节码文件的 html 文件，从而获得运行结果。
8. Java 语言具有较好的安全性和可移植性及与平台无关等特性。( )
9. Java 语言的基本数据类型有 4 种：整型、浮点型、字符型和布尔型。( )



- 10.子类所继承父类的成员都可以在子类中访问。 ( )
- 11.Java 语言规定在嵌套的程序块中不允许定义同名的成员变量。 ( )
- 12.Java 语言采用 16 位颜色标准。Java 的调色板保证 128 色。
- 13.静态方法只能处理静态变量。 ( )
- 14.画布 (Canvas) 是一种可容纳多个组件的容器。 ( )
- 15.Java 语言是一种强类型语言, 数据类型转换有两种: 隐含转换和强制转换。 ( )
- 16.Java 语言中, 数组在静态和动态赋值时都判越界。 ( )
- 17.Frame 容器是有边框的容器, 它也是一种独立窗口, 只能作为最外层容器。 ( )
- 18.集合 Set 是通过键-值对的方式来存储对象的。
- 19.异常处理是在编译时进行的。 ( )
- 20.死锁的产生是因为多个线程间存在资源竞争。 ( )

#### 四、阅读下列程序, 请写出程序的输出结果

1.

```
class B {
    int b;

    B(int x) {
        b = x;
        System.out.println("b=" + b);
    }
}

class A extends B {
    int a;

    A(int x, int y) {
        super(x);
        a = y;
        System.out.println("b=" + b + ", a=" + a);
    }
}

public class Test {
```

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

```
    A obj = new A(13, 23);
```

```
}
```

```
}
```

答案

b=13

b=13, a=23

2.

```
public class Test1 {
```

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

```
        AB s = new AB("Hello!", "I love JAVA.");
```

```
        System.out.println(s.toString());
```

```
    }
```

```
}
```

```
class AB {
```

```
    String s1;
```

```
    String s2;
```

```
    AB(String str1, String str2) {
```

```
        s1 = str1;
```

```
        s2 = str2;
```

```
    }
```

```
    public String toString() {
```

```
        return s1 + s2;
```

```
    }
```

```
}
```

答案 Hello! I love JAVA.

3. public class Test {

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

```
        int[][] num1 = new int[2][3];
```

```
        num1[0][0] = 3;
```

```

num1[0][1] = 9;
num1[0][2] = 60;
num1[1][0] = 78;
num1[1][1] = 79;
for (int i = 0; i <= 1; i++) {
    for (int j = 0; j < num1[i].length; j++)
        System.out.print(num1[i][j] + "\t");
    System.out.println();
}
}

```

答案

```

3  9  60
78 79 0

```

```

4. public class Test {
    public static void main(String[] args) {
        int i, s = 0;
        int a[] = { 10, 20, 30, 40, 50, 60, 70, 80, 90 };
        for (i = 0; i < a.length; i++)
            if (a[i] % 3 == 0) {
                s += a[i];
                System.out.println("s=" + s);
            }
    }
}

```

答案

```

s=30
s=90
s=180

```

```

5. public class Test3 {

```

```

public static void main(String[] args) {
    A a = new A(2);        A b = new A(3);
    System.out.println(a.isEqualTo(b));
}
}

class A {
    private int privateVar;
    public A(int _privateVar) {        privateVar = _privateVar;    }
    boolean isEqualTo(A anotherA) {
        if (this.privateVar == anotherA.privateVar)        return true;
        else        return false;
    }
}

```

答案

false

```

6. public class Test {
    public static void main(String[] args) {
        int[][] num1 = new int[2][3];
        num1[0][0] = 3;        num1[0][1] = 9;        num1[0][2] = 45;
        num1[1][0] = 78;        num1[1][1] = 79;
        for (int i = 0; i <= 1; i++) {
            for (int j = 0; j < num1[i].length; j++)
                System.out.print(num1[i][j] + " ");
            System.out.println();
        }
        int[][] num2 = new int[][] { { 1, 2, 3 }, { 7, 8 }, { 9 } };
        for (int i = 0; i < num2.length; i++) {
            for (int j = 0; j < num2[i].length; j++)
                System.out.print(num2[i][j] + " ");
        }
    }
}

```

```
System.out.println();
```

```
}
```

```
}
```

```
}
```

答案

3 9 45

78 79 0

1 2 3

7 8

9

7. public class Test2 {

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

```
    int i, s = 0;
```

```
    int a[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9 };
```

```
    for (i = 0; i < a.length; i++)
```

```
        if (a[i] % 3 == 0) {
```

```
            s += a[i];
```

```
            System.out.println("s=" + s);
```

```
        }
```

```
    }
```

```
}
```

答案

s=3

s=9

s=18

8. class B {

```
    int b;
```

```
    B(int x) {
```

```
        b = x;
```

```
        System.out.println("b=" + b);
```

```
    }
```

```

    }

    class A extends B {

        int a;

        A(int x, int y) {
            super(x);

            a = y;

            System.out.println("b=" + b + ", a=" + a);
        }
    }

    public class Test {

        public static void main(String[] args) {

            A obj = new A(5, 10);

        }
    }

```

答案

b=5

b=5, a=10

```

9. public class Test4 {

    public static void main(String[] args) {

        try {

            method();

        } catch (Exception e) {

            System.out.println("A");

        } finally {

            System.out.println("B");

        }

    }

    static void method() {

        try {

            wrench();

```

```

        System.out.println("C");
    } catch (ArithmeticException e) {
        System.out.println("D");
    } finally {
        System.out.println("E");
    }
    System.out.println("F");
}

static void wrench() {
    throw new NullPointerException();
}
}

```

答案

E  
A  
B

## 五、阅读程序，回答问题

1.

```

1: public class Output1 {
2:     public static void main(String arge[]) {
3:         int i=0;
4:         for ( char ch = 97; ch<113; ch++,i++) {
5:             if( i % 8 == 0 )
6:                 System.out.println(" ");
7:                 System.out.print("\t" +ch);
8:         }
9:     }
10: }

```

- (1) 程序第 5、6 行的 if 语句的功能是什么？
- (2) 程序输出的结果有几行？

2.

```
1: import java.util.Arrays;
2: public class SortArray {
3:     public static void main(String args[]) {
4:         String[] str = {"size", "abs", "length", "class"};
5:         Arrays.sort(str);
6:         for (int i=0; i<str.length; i++)
7:             System.out.print(str[i]+" ");
8:     }
9: }
```

(1) 写出程序运行后的结果。

(2) 在第 4 行和 5 行之间, 进行 str[0].length()方法调用的返回结果是多少?

3.

```
abstract class SuperAbstract{
    void a(){...}
    abstract void b();
    abstract int c(int i);
}

interface AsSuper{
    void x();
}

abstract class SubAbstract extends SuperAbstract implements AsSuper{
    public void b(){...}
    abstract String f();
}

public class InheritAbstract extends SubAbstract{
    public void x(){...}
    public int c(int i) {...}
    public String f(){...}
    public static void main(String args[]){
```



```

        InheritAbstract instance=new InheritAbstract();

        instance.x();

        instance.a();

        instance.b();

        instance.c(100);

        System.out.println(instance.f());

    }

}

```

根据以上这段程序，回答问题：

- (1) 哪几个类是抽象类，写出类名。
- (2) 哪几个类是非抽象类，写出类名。
- (3) 是否有接口，写出接口名。

4. 定义类 A 和类 B 如下。

```

class A{

    int a=1;

    double d=2.0;

    void show() {

        System.out.println("Class A: a="+a +"\td="+d);

    }

}

class B extends A{

    float a=3.0f;

    String d="Java program.";

    void show() {

        super.show();

        System.out.println("Class B: a="+a +"\td="+d);

    }

}

```

若在应用程序的 main 方法中有以下语句：

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

```
a.show();
```

则输出的结果如何? Class A: a=1 d=2.0

5. 写出此程序完成的功能。

```
public class Sum {  
    public static void main(String args[]) {  
        double sum = 0.0;  
        for (int i = 1; i <= 100; i++)  
            sum += 1.0 / (double) i;  
        System.out.println("sum=" + sum);  
    }  
}
```

回答问题：以上程序完成什么功能？

运行结果是：sum=5.187377517639621

6. 写出此程序完成的功能。

```
import java.io.*;  
public class ArrayCount {  
    public static void main(String[] args) {  
        int i, n = 10, sum = 0, count = 0;  
        int a[] = new int[10];  
        for (i = 0; i < n; i++)  
            try {  
                BufferedReader br = new BufferedReader(new  
                    InputStreamReader(System.in));  
                a[i] = Integer.parseInt(br.readLine());  
            } catch (IOException e) {  
            }  
        for (i = 0; i < n; i++)  
            sum += a[i];  
    }  
}
```

```

        sum /= n;

        for (i = 0; i < n; i++)
            if (a[i] < sum)
                ++count;

        System.out.println(count);
    }
}

```

7.

```

import java.util.*;

public class Test4{

    public static void main(String args[]) {

        Set set = new TreeSet();

        set.add(new Integer(10));

        set.add(new Integer(5));

        set.add(new Integer(15));

        set.add(new Integer(5));

        set.add(new Integer(10));

        System.out.println("size = " + set.size());

        Iterator it = set.iterator();

        while (it.hasNext()) {

            System.out.print(it.next() + " ");

        }

    }
}

```

(1) 程序运行后输出的结果如何?

size = 3

5 10 15

(2) 说明 java 中的集合(Set 接口)和映射(Map 接口)的主要区别。

8.

1: import java.applet.Applet;

```

2:  import java.awt.*;
3:  import java.awt.event.*;
4:  public class TestKeyListener extends Applet {
5:      Color clr;
6:      public void init() {
7:          addKeyListener(
8:              new KeyAdapter() {
9:                  public void keyTyped( KeyEvent e ) {
10:                     if(e.getKeyChar()=='r' || e.getKeyChar()=='R')
11:                         clr=Color.red;
12:                     else if(e.getKeyChar()=='g' || e.getKeyChar()=='G')
13:                         clr=Color.green;
14:                     else if(e.getKeyChar()=='b' || e.getKeyChar()=='B')
15:                         clr=Color.blue;
16:                     else
17:                         clr=Color.black;
18:                     setBackground(clr);
19:                 }
20:             });
21:     }
22: }

```

(1) 程序第 10、11 行实现的功能是什么？返回引发键盘事件的按键所对应的 Unicode 字符，如果为大写 R 或小写 r 则让变量 clr 等于 Color 类 red 方法。

(2) 编译该程序，并编写嵌入该程序字节码文件的网页后，通过支持 java 的浏览器打开网页。如果敲击键盘的'b'键，有什么结果？

## 六、程序设计题

1. 设计一个 Calculator 类，并完成下列的各方法的程序设计并测试：

(1) 定义 add()方法，用来计算两数(a,b)之和。

(2) 定义 sub()方法，用来计算两数(a,b)之差。

(3) 定义 mul() 方法, 用来计算两数(a,b)之积。

(4) 定义 div() 方法, 用来计算 a/b。

```
public class kk {

    int a, b;

    int add() {
        return (a + b);
    }

    int sub() {
        return (a - b);
    }

    int mul() {
        return (a * b);
    }

    int div() {
        if (b != 0)
            return (a / b);
        else
            return 0;
    }

    public static void main(String[] args) {
        kk rect = new kk();
        rect.a = 10;
        rect.b = 6;
        System.out.println("add=" + rect.add());
        System.out.println("sub=" + rect.sub());
        System.out.println("mul=" + rect.mul());
        System.out.println("div=" + rect.div());
    }
}
```

2. 定义一个 ArrayList 集合对象, 往其中添加随机产生的 0-10 之间的 float 类型数据, 共添加 10 个, 并打印输出该集合对象中每个元素的值。

```
import java.util.ArrayList;

import java.util.Iterator;

public class Test {

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

```

ArrayList list = new ArrayList();

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

    list.add(i, (float) Math.random()*10);

for(int j=0;j<list.size();j++)

System.out.println(list.get(j));

    Iterator it=list.iterator();

while(it.hasNext()){

    System.out.println(it.next());

}

}

}

```

3. 试利用 do while 循环, 计算 10!。

```

public class Sum {

    public static void main(String[] args) {
        int s = 1;
        for(int i=1;i<=10;i++)
        {
            s*=i;
        }
        System.out.println(s);
    }

}

```

其他问题:

1、用输入流读、并统计用户输入信息中的字母、数字

import java.io.\*;

public class Test11 {

public static void main(String[] args) throws IOException {

int nDig=0,nChar=0,nOther=0;

InputStreamReader in=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(in);

System.out.println("请输入: ");

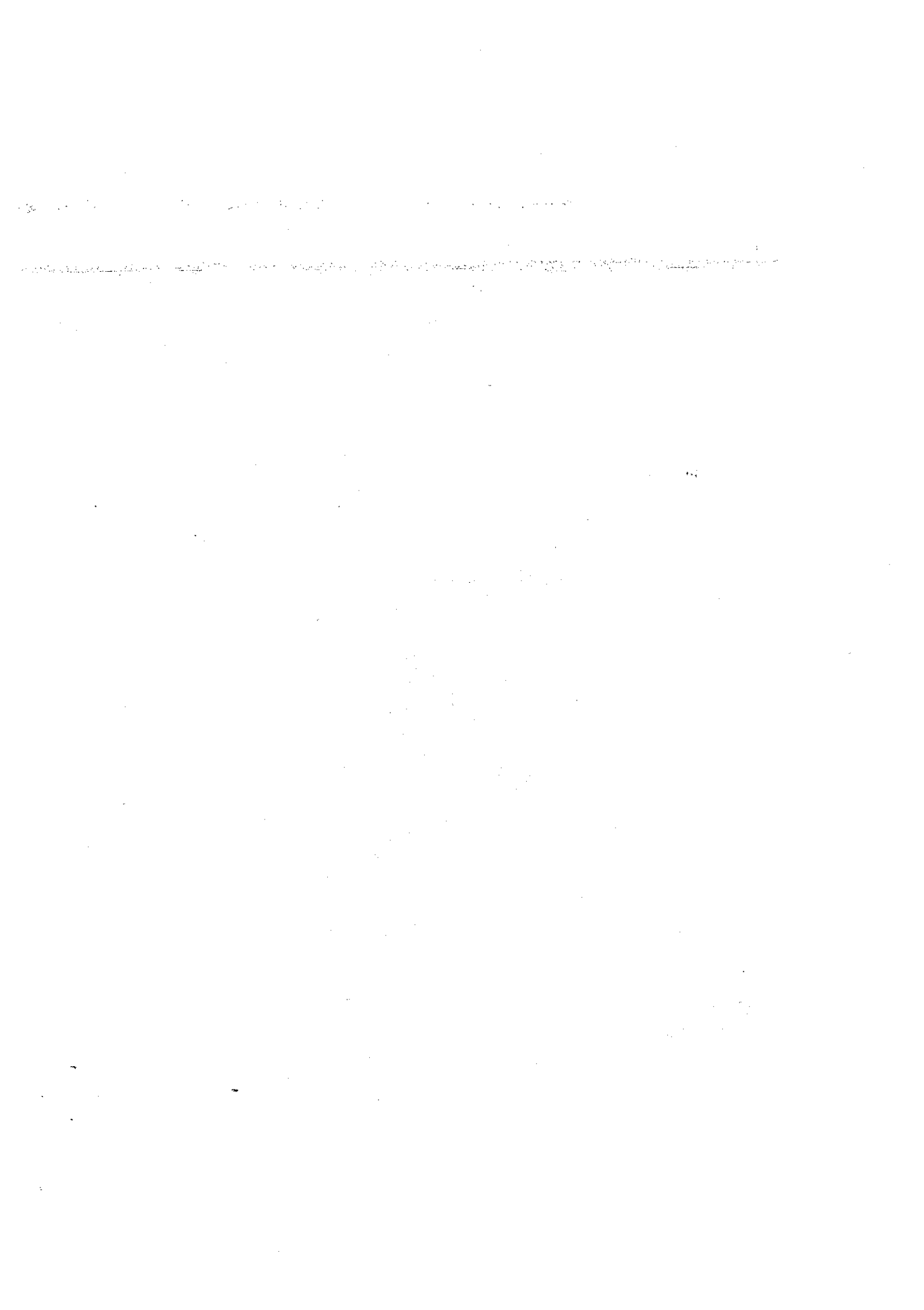
String str=br.readLine();

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

```
        if(str.charAt(i)>='0'&&str.charAt(i)<='9'){
            nDig++;
        } else if(str.charAt(i)>='a'&&str.charAt(i)<='z' || str.charAt(i)>='A'&&str.charAt(i)<='Z'){
            nChar++;
        } else
            nOther++;
    }

    System.out.println("字母: "+nChar);
    System.out.println("数字: "+nDig);
    System.out.println("其它字符: "+nOther);
}
}
```

2、循环语句的使用





# 《JAVA 语言程序设计》期末考试试题及答案 1

## 一、单选题

1、编译 Java Application 源程序文件将产生相应的字节码文件，这些字节码文件的扩展名为( )。

- A. java
- B. .class
- C. html
- D. .exe

2、设  $x=1, y=2, z=3$ ，则表达式  $y+=z--/++x$  的值是( )。

- A. 3
- B. 3.5
- C. 4
- D. 5

3、不允许作为类及类成员的访问控制符的是( )。

- A. public
- B. private
- C. static
- D. protected

4、为 AB 类的一个无形式参数无返回值的方法 method 书写方法头，使得使用类名 AB 作为前缀就可以调用它，该方法头的形式为( )。

- A. static void method()
- B. public void method()
- C. final void method()
- D. abstract void method()

## 二、填空题

1、开发与运行 Java 程序需要经过的三个主要步骤为 编辑源程序、编译生成字节码 和 解释运行字节码。

2、在 Java 的基本数据类型中，char 型采用 Unicode 编码方案，每个 Unicode 码占 2 字节内存空间，这样，无论是中文字符还是英文字符，都是占用 2 字节内存空间。

3、设  $x=2$ ，则表达式  $(x++)/3$  的值是 0。

4、若  $x=5, y=10$ ，则  $x < y$  和  $x \geq y$  的逻辑值分别为 true 和 false。

5、抽象(abstract) 方法是一种仅有方法头，没有具体方法体和操作实现的方法，该方法必须在抽象类之中定义。最终(final) 方法是不能被当前类的子类重新定义的方法。

6、创建一个名为 MyPackage 的包的语句是 package MyPackage;，该语句应该放在程序的位置为：应该在程序第一句。

7、设有数组定义：int MyIntArray[] = { 10, 20, 30, 40, 50, 60, 70}; 则执行以下几个语句后的输出结果是 120。

int s = 0;

```
for (int i = 0; i < MyIntArray.length; i++)
```

```
    if (i % 2 == 1)
```

```
        s += MyIntArray[i];
```

```
System.out.println(s);
```

8、在 Java 程序中，通过类的定义只能实现单重继承，但通过接口的定义可以实现多重继承关系。

三、写出下面程序的运行结果

1、 import java.io.\*;

```
public class abc
```

```
{
```

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

```
    {
```

```
        AB s = new AB("Hello!", "I love JAVA.");
```

```
        System.out.println(s.toString());
```

```
    }
```

```
}
```

```
class AB {
```

```
    String s1;
```

```
    String s2;
```

```
    public AB(String str1, String str2)
```

```
    {
```

```
        s1 = str1;
```

```
        s2 = str2;
```

```
    }
```

```
    public String toString( )
```

```
    {
```

```
        return s1+s2;
```

```
    }
```

```
}
```

运行结果：Hello! I love JAVA.

2、 import java.io.\*;

```
public class abc
```

```

{
    public static void main(String args[ ])
    {
        int i, s = 0;
        int a[ ] = { 10, 20, 30, 40, 50, 60, 70, 80, 90 };
        for ( i = 0 ; i < a.length ; i ++ )
            if ( a[i]%3 == 0 )    s += a[i] ;
        System.out.println("s="+s);
    }
}

```

运行结果: s = 180

```

3、 import java.io.* ;
   public class abc
   {
       public static void main(String args[ ])
       {
           System.out.println("a="+a+"\nb="+b);
       }
   }
   class SubClass extends SuperClass
   {   int c;
       SubClass(int aa, int bb, int cc)
       {
           super(aa, bb);
           c=cc;
       }
   }
   class SubSubClass extends SubClass
   {   int a;
       SubSubClass(int aa, int bb, int cc)
       {   super(aa, bb, cc);
           A = aa+bb+cc;
       }
       void show()

```

```

    {
        System.out.println("a="+a+"\nb="+b+"\nc="+c);
    }
}

```

运行结果: a=60

b=20

c=30

## 《JAVA 语言程序设计》期末考试试题及答案 2 (应考必备题库)

### 一. 判断题

1. Java 的源代码中定义几个类, 编译结果就生成几个以.class 为后缀的字节码文件。(√)
2. Java 程序里, 创建新的类对象用关键字 new, 回收无用的类对象使用关键字 free。(×)
3. Java 有垃圾回收机制, 内存回收程序可在指定的时间释放内存对象。(×)
4. 构造函数用于创建类的实例对象, 构造函数名应与类名相同, 返回类型为 void。(×)
5. 在异常处理中, 若 try 中的代码可能产生多种异常则可以对应多个 catch 语句, 若 catch 中的参数类型有父类子类关系, 此时应该将父类放在后面, 子类放在前面。(√)
6. 拥有 abstract 方法的类是抽象类, 但抽象类中可以没有 abstract 方法。(√)
7. Java 的屏幕坐标是以像素为单位, 容器的左下角被确定为坐标的起点。(×)
8. 静态初始化器是在其所属的类加载内存时由系统自动调用执行。(√)
9. 在 Java 中对象可以赋值, 只要使用赋值号(等号)即可, 相当于生成了一个各属性与赋值对象相同的新对象。(×)

### 二. 单项选择题

1. Java application 中的主类需包含 main 方法, 以下哪项是 main 方法的正确形

参? ( )

A、String args    B、String ar[]    C、Char arg    D、StringBuffer args[]

2. 以下关于继承的叙述正确的是 ( )。

A、在 Java 中类只允许单一继承

B、在 Java 中一个类只能实现一个接口

C、在 Java 中一个类不能同时继承一个类和实现一个接口

D、在 Java 中接口只允许单一继承

3. paint()方法使用哪种类型的参数? ( )

A、Graphics    B、Graphics2D    C、String    D、Color

4. 以下哪个不是 Java 的原始数据类型 ( )

A、int    B、Boolean    C、float    D、char

5. 以下哪项可能包含菜单条 ( )。

A、Panel    B、Frame    C、Applet    D、Dialog

6. 若需要定义一个类域或类方法, 应使用哪种修饰符? ( )

A、static    B、package    C、private    D、public

7. 在浏览器中执行 applet 程序, 以下选项中的哪个方法将被最先执行 ( )。

A、init()    B、start()    C、destroy()    D、stop()

8. 给出下面代码, 关于该程序以下哪个说法是正确的? ( )

```
public class Person{  
    static int arr[] = new int[5];  
    public static void main(String a[])  
    {  
        System.out.println(arr[0]);  
    }  
}
```

A、编译时将产生错误

B、编译时正确, 运行时将产生错误

C、输出零

D、输出空

9. 下列哪些语句关于 Java 内存回收的说明是正确的? ( )

A、程序员必须创建一个线程来释放内存

B、内存回收程序负责释放无用内存

C、内存回收程序允许程序员直接释放内存

D、内存回收程序可以在指定的时间释放内存对象

10. 以下哪个关键字可以用来对对象加互斥锁? ( )

A、transient      B、synchronized      C、serialize      D、static

11. 以下代码段执行后的输出结果为 ( )

```
int x=-3;
int y=-10;
System.out.println(y%x);
```

A、-1      B、2      C、1      D、3

12. 有以下程序片段，下列哪个选项不能插入到行1。( )

1.

2. public class Interesting{

3. //do sth

4.}

A、import java.awt.\*;      B、package mypackage;  
C、class OtherClass{ }      D、public class MyClass{ }

13. 设有下面两个赋值语句:

```
a = Integer.parseInt("12");
b = Integer.valueOf("12").intValue();
```

下述说法正确的是 ( )。

A、a 是整数类型变量，b 是整数类对象。

B、a 是整数类对象，b 是整数类型变量。

C、a 和 b 都是整数类对象并且值相等。

D、a 和 b 都是整数类型变量并且值相等。

14. FilterOutputStream 是 BufferedOutputStream、DataOutputStream 及 PrintStream 的父类，以下哪个类可能是 FilterOutputStream 构造函数的参数类型?

A、OutputStream      B、File  
C、InputStream      D、BufferedOutputStream

15. 在编写 Java Applet 程序时，需在程序的开头写上( )语句。

A、import java.awt.\*;      B、import java.applet.Applet;

C、import java.io.\*;

D、import java.awt.Graphics;

16. 类 Parent、Child 定义如下:

```
1. public class Parent
2. { public float aFun(float a, float b) throws
3.   IOException {
4. }
5. public class Child extends Parent{
6.
7. }
```

将以下哪种方法插入行 6 是不合法的。( )

A、float aFun(float a, float b){ }

B、public int aFun(int a, int b) throws Exception{ }

C、public float aFun(float p, float q){ }

D、public int aFun(int a, int b) throws IOException{ }

17. 在使用 interface 声明一个接口时, 只可以使用 ( ) 修饰符修饰该接口。

A、private

B、protected

C、private protected

D、public

18. 有以下方法的定义, 请选择该方法的返回类型 ( )。

```
ReturnType method(byte x, double y)
{
    return (short) x/y*2;
}
```

A、byte

B、short

C、int

D、double

19. 关于以下代码段的说法正确的是 ( )

```
1. String s="abcde";
2. StringBuffer s1=new StringBuffer("abcde");
3. if(s.equals(s1))
4.     s1=null;
5. if(s1.equals(s))
6.     s=null;
```

A、第 1 行编译错误, String 的构造器必须明确调用

B、第 3 行编译错误, 因为 s 与 s1 有不同的类型

C、编译成功，但执行时在第 5 行有异常抛出

D、编译成功，过程中也没有异常抛出

20、编译并运行以下程序，以下描述哪个选项是正确的 ( )

```
1. class X{
2.     protected String toString() {
3.         return super.toString();
4.     }
```

A、编译通过运行无异常

B、编译通过但运行时出错

C、行 2 出错，不能成功编译

D、不能成功编译，行 3 出错

### 三、程序阅读题

1. 以下程序的输出结果为\_\_\_\_相等\_\_\_\_。

```
class StringTest1
{
    public static void main(String[] args)
    {
        String s1="hello";
        String s2=new String("hello");
        if(s1.equals(s2)){
            System.out.println("相等");
        }else{
            System.out.println("不相等");
        }
    }
}
```

2. 以下程序段的输出结果为\_\_\_\_5 6 7 8 9\_\_\_\_。

```
public class TestArray
{
    public static void main(String args[]){
        int i,j;
        int a[] = { 5,9,6,8,7};
        for ( i = 0 ; i < a.length-1; i ++ ) {
```



```

        int k = i;
        for (j = i; j < a.length; j++)
            if (a[j] < a[k]) k = j;
        int temp = a[i];
        a[i] = a[k];
        a[k] = temp;
    }
    for (i = 0; i < a.length; i++)
        System.out.print(a[i] + " ");
    System.out.println();
}
}

```

3. 写出以下程序的功能。

```

import java.io.*;
public class TestFile
{
    public static void main(String args[]) throws Exception
    {
        BufferedReader br = new BufferedReader(
            new InputStreamReader(System.in));
        BufferedWriter bw = new BufferedWriter(new FileWriter("input.txt"));
        String s;
        while (true)
        {
            System.out.print("请输入一个字符串: ");
            System.out.flush();
            s = br.readLine();
            if (s.length() == 0) break;
            bw.write(s);
            bw.newLine();
        }
        bw.close();
    }
}

```

}

功能：从键盘逐一输入字符串，逐一输出至 input.txt 文件中，直至输入空行。

4. 阅读以下程序，写出输出结果。

```
class Animal {  
    Animal() {  
        System.out.print ("Animal "); }  
}  
  
public class Dog extends Animal {  
    Dog() {  
        System.out.print ("Dog "); }  
  
    public static void main(String[] args) {  
        Dog snoppy= new Dog(); }  
}
```

输出结果：Animal Dog

## 《JAVA 语言程序设计》期末考试试题及答案 3（应考必备题库）

### 一、单项选择题

1、如下哪个是 Java 中的标识符（）

A、public      B、super      C、3number      D、width

2、如下哪个是 Java 中的标识符（）

A、fieldname      B、super      C、3number      D、#number

3、已知如下定义：String s = "story"; 下面哪个语句不是合法的（）

A、s += "books";      B、s = s + 100;

C、int len = s.length;

D、String t = s + "abc";

4、如下哪个是 Java 中有效的关键字()

A、name

B、hello

C、false

D、good

5、下面的代码段执行之后 count 的值是什么( )

```
int count = 1;
for (int i = 1; i <= 5; i++) {
    count += i;
}
```

System.out.println(count);

A、5

B、1

C、15

D、16

6、定义一个类，必须使用的关键字是( )

A、public

B、class

C、interface

D、static

7、定义一个接口必须使用的关键字是( )

A、public

B、class

C、interface

D、static

8、如果容器组件 p 的布局是 BorderLayout，则在 p 的下边中添加一个按钮 b，应该使用的语句是( )

A、p.add(b);

B、p.add(b,"North");

C、p.add(b,"South");

D、b.add(p,"North");

9、声明并创建一个按钮对象 b，应该使用的语句是( )

A、Button b=new Button();

B、button b=new button();

C、Button b=new b();

D、b.setLabel("确定");

10、Frame 对象默认的布局管理器是( )

A、FlowLayout

B、BorderLayout

C、CardLayout

D、null

11、下列哪一个 import 命令可以使我们在程序中创建输入/输出流对象 ( )

A、import java.sql.\*;

B、import java.util.\*;

C、import java.io.\*;

D、import java.net.\*;

12、下面哪一个 import 命令可以为我们提供编写网络应用程序的类 ( )

- A、import java.sql.\*;      B、import java.util.\*;  
C、import java.io.\*;      D、import java.net.\*;

13、如果需要从文件中读取数据，则可以在程序中创建哪一个类的对象 ( )

- A、FileInputStream      B、FileOutputStream  
C、DataOutputStream      D、FileWriter

## 二、填空题

1、如果将类 MyClass 声明为 public，它的文件名称必须是 (MyClass.java) 才能正常编译。

2、Java 程序中的单行注释符是 ( // )，多行注释符是 ( /\* \*/ )。

3、Java 中布尔类型的常量有两种，它们是 (true) 和 (false)。

4、Java 中用于定义小数的关键字有两个：(float) 和 (double)，后者精度高于前者。

5、Java 中用于两个数相等比较的运算符是：(==)，用于不相等比较的运算符是 (<>)。

6、在 Java 中定义一个字符串类型的变量 str 的语句是：(String str;)，定义一个具有 10 个元素的整型数组 a 的语句是：(int[] arr = new int[10];)。

7、导入 mypackage 包中的所类的命令是 ( import mypackage.\*; )。

8、当声明一个数组 int arr[] = new int[5]; 时，这代表这个数组所保存的变量类型是 (int)，数组名是 (arr)，数组的大小为 (5)，数组元素下标的使用范围是 (0 到 4)。

9、假设 x=13, y=4，则表达式 x%y != 0 的值是 (true)，其数据类型是 (boolean)。

10、异常处理是由 (try)、(catch) 和 finally 块三个关键所组成的程序块。

11、以下程序段的输出结果是 (三角形)

```

int x = 5, y = 6, z = 4;
if (x + y > z && x + z > y && z + y > x)
    System.out.println("三角形");
else
    System.out.println("不是三角形");

```

12、下面程序段的执行结果是 (65432)

```

int a[] = { 2, 3, 4, 5, 6 };
for (int i = a.length - 1; i >= 0; i--)
    System.out.print(a[i] + " ");

```

### 三、程序阅读题

1、以下程序的输出结果为 Peter is 17 years old!。

```

public class Person {
    String name;
    int age;

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public static void main(String[] args) {
        Person c = new Person("Peter", 17);
        System.out.println(c.name + " is " + c.age + " years old!");
    }
}

```

2、以下程序的输出结果为 课程号:101 课程名:ASP 学分:3。

```

public class Course {
    private String cNumber;
    private String cName;
    private int cUnit;

    public Course(String number, String name, int unit) {
        cNumber = number;
        cName = name;
        cUnit = unit;
    }

    public void printCourseInfo() {
        System.out.println("课程号:" + cNumber + " 课程名:" + cName + " 学

```

```
分:" + cUnit);
```

```
}
```

```
}
```

```
class CourseTest {
```

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

```
        Course c;
```

```
        c = new Course("101", "ASP", 3);
```

```
        c.printCourseInfo();
```

```
    }
```

```
}
```

3、以下程序的输出结果为\_\_汤姆猫体重: 20.0 斤\_\_。

```
public class Tom {
```

```
    private float weight;
```

```
    private static String name;
```

```
    public void setWeight(float weight) {
```

```
        this.weight = weight;
```

```
    }
```

```
    private void out() {
```

```
        System.out.println(name + "体重: " + weight + "斤");
```

```
    }
```

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

```
        Tom.name = "汤姆猫";
```

```
        Tom cat = new Tom();
```

```
        cat.setWeight(20);
```

```
        cat.out();
```

```
    }
```

```
}
```

4、以下程序的输出结果\_姓名:Tom 年龄:15 家庭住址:金水区 电话:66123456  
学校: 九中\_。

```
public class Father {
```

```
    String name, address, tel;
```

```
    int age;
```

```
    public Father(String name, int age) {
```

```
        this.name = name;
```

```
        this.age = age;
```

```

    }

    void out() {
        System.out.print("姓名:" + name);
        System.out.print(" 年龄:" + age);
    }

    void outOther() {
        System.out.print(" 家庭住址:" + address);
        System.out.print(" 电话:" + tel);
    }
}

class Son extends Father {
    String school;

    public Son(String name, int age) {
        super(name, age);
    }

    void out() {
        super.out();
        super.outOther();
        System.out.println(" 学校: " + school);
    }

    public static void main(String args[]) {
        Son son = new Son("Tom", 15);
        son.address = "金水区";
        son.school = "九中";
        son.tel = "66123456";
        son.out();
    }
}

```

5、下列程序的运行结果是\_\_12345\_\_。

```

public class MyClass {
    int a[] = { 1, 2, 3, 4, 5 };

    void out() {
        for (int j = 0; j < a.length; j++)
            System.out.print(a[j] + "");
    }
}

```

```

    public static void main(String[] args) {
        MyClass my = new MyClass();
        my.out();
    }
}

```

#### 四、简答题

##### 1、简述 Java 中异常处理的机制？

答：首先 Java 的异常是面向对象的。一个 Java 的 Exception 是一个描述异常情况的对象。当出现异常情况时，一个 Exception 对象就产生了，并放到异常的成员函数里。

Java 的异常处理是通过 5 个关键词来实现的：try, catch, throw, throws 和 finally。

1. 异常处理：在 Java 语言的错误处理结构由 try, catch, finally 三个块组成。其中 try 块存放将可能发生异常的 Java 语言，并管理相关的异常指针；catch 块紧跟在 try 块后面，用来激发被捕获的异常；finally 块包含清除程序没有释放的资源，句柄等。不管 try 块中的代码如何退出，都将执行 finally 块。

2. 抛出异常：Java 语言可以不在方法中直接捕获，而用 throw 语句将异常抛给上层的调用者。Throw 语句就是来明确地抛出一个异常：首先你必需得到一个 Throwable 的实例句柄，通过参数传到 catch 中，或者采用 new 操作符来创建一个。

##### 2、什么是继承？

答：通过必要的说明能够实现某个类无需重新定义就拥有另一个类的某些属性和方法，并把这种关系称为继承，先定义的类称为父类，后定义的类称为子类，并且允许多层的继承关系。

##### 3、请简述重载和重写的区别？

答：方法的重写 Overriding 和重载 Overloading 是 Java 多态性的不同表现。

重写 Overriding 是父类与子类之间多态性的一种表现，重载 Overloading 是一个类中多态性的一种表现。

如果在子类中定义某方法与其父类有相同的名称和参数，我们说该方法被重写 (Overriding)。子类的对象使用这个方法时，将调用子类中的定义，对它而言，父类中的定义如同被“屏蔽”了。

如果在一个类中定义了多个同名的方法，它们或有不同的参数个数或有不同的参数类型，则称为方法的重载 (Overloading)。Overloaded 的方法是可以改变返回值的类型。



## 五、程序设计题

### 1、按以下要求编写程序

- (1) 创建一个 Rectangle 类, 添加 width 和 height 两个成员变量
- (2) 在 Rectangle 中添加两种方法分别计算矩形的周长和面积
- (3) 编程利用 Rectangle 输出一个矩形的周长和面积

解答:

```
public class Rectangle {
    float width, height;

    public Rectangle(float width, float height) {
        this.width = width;
        this.height = height;
    }

    public float getLength(){
        return (this.width + this.height) * 2;
    }

    public float getArea(){
        return this.width * this.height;
    }

    public static void main(String [] args) {
        Rectangle rect = new Rectangle(10, 20);
        System.out.println("周长是: " + rect.getLength());
        System.out.println("面积是: " + rect.getArea());
    }
}
```

### 2、按以下要求编写程序

- (1) 编写 Animal 接口, 接口中声明 run() 方法
- (2) 定义 Bird 类和 Fish 类实现 Animal 接口
- (3) 编写 Bird 类和 Fish 类的测试程序, 并调用其中的 run() 方法

解答:

```
public interface Animal {
    void run();
}

class Bird implements Animal {
    public void run() {
        System.out.println("鸟儿在飞...");
    }
}
```

```

}

class Fish implements Animal {
    public void run() {
        System.out.println("鱼儿在游...");
    }
}

public class TestAnimal {

    public static void main(String[] args) {
        Bird bird = new Bird();
        bird.run();

        Fish fish = new Fish();
        fish.run();
    }
}

```

3、编写一个 Frame 框架应用程序，要求如下：

- (1) 在窗口设置两个菜单“文件”、“编辑”
- (2) 在“文件”菜单里添加三个菜单项“打开”、“保存”、“关闭”
- (3) 在“编辑”菜单里添加两个菜单项“复制”、“粘贴”
- (4) 点击关闭菜单项时，使程序关闭。

解答：

```

import java.awt.Color;
import java.awt.Frame;
import java.awt.Menu;
import java.awt.MenuBar;
import java.awt.MenuItem;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

/*
 * 主窗体
 */
public class MyFrame extends Frame {
    private static final long serialVersionUID = 6895463895656626294L;

    private Frame frmMain;    // 窗体

```

```

MenuBar mb;                // 菜单栏
Menu mnuFile;              // "文件"菜单
Menu mnuEdit;              // "编辑"菜单
MenuItem miOpen;           // "打开"菜单项
MenuItem miSave;           // "保存"菜单项
MenuItem miClose;          // "关闭"菜单项
MenuItem miCopy;           // "复制"菜单项
MenuItem miPaste;          // "粘贴"菜单项

public MyFrame() {
    frmMain = new Frame("主窗体");

    // 创建菜单栏
    mb = new MenuBar();

    // 创建菜单
    mnuFile = new Menu("文件");
    mnuEdit = new Menu("编辑");

    // 创建菜单项
    miOpen = new MenuItem("打开");
    miSave = new MenuItem("保存");
    miClose = new MenuItem("关闭");
    miCopy = new MenuItem("复制");
    miPaste = new MenuItem("粘贴");
}

public void showFrame() {
    frmMain.setSize(800, 600);
    frmMain.setLocation(100, 100);
    frmMain.setBackground(Color.white);
    frmMain.setVisible(true);
    frmMain.setLayout(null);

    frmMain.addWindowListener(new WindowHandler()); // 注册窗口监听器

    // 将菜单栏放入窗体中
    frmMain.setMenuBar(mb);

    // 将菜单放入菜单栏中
    mb.add(mnuFile);
    mb.add(mnuEdit);

    // 将菜单项放入菜单中

```

```

        mnuFile.add(miOpen);
        mnuFile.add(miSave);
        mnuFile.add(miClose);

        mnuEdit.add(miCopy);
        mnuEdit.add(miPaste);

        // 注册动作事件监听器
        miClose.setActionCommand("miClose_Clicked");
        miClose.addActionListener(new MenuHandler());
    }

    // 定义一个内部类，在这个类中编写窗口关闭事件处理程序。
    private class WindowHandler extends WindowAdapter {

        public void windowClosing(WindowEvent e) {
            System.exit(0); // 退出应用程序，状态码 0 表示正常终止，非 0 表示异常终止。
        }
    }

    // 定义一个内部类，在这个类中编写动作事件处理程序。
    private class MenuHandler implements ActionListener {

        public void actionPerformed(ActionEvent e) {

            if (e.getActionCommand() == "miClose_Clicked") {
                System.exit(0); // 退出应用程序
            }
        }
    }
}

public class TestMyFrame {

    public static void main(String[] args) {
        // 启动主窗体
        MyFrame guiWindow = new MyFrame();
        guiWindow.showFrame();
    }
}

```

## 《JAVA 语言程序设计》期末考试试题及答案 4 (应考必备题库)

### 一、单选题

1、下列程序段执行后的结果是( )。

```
String s = new String("abcdefg");
for (int i=0; i<s.length(); i+=2){
    System.out.print(s.charAt(i));
}
```

A) aceg    B) ACEG    C) abcdefg    D) abcd

2、有整型数组: int[] x={12,35,8,7,2};,则调用方法 Arrays.sort(x)后, 数组 x 中的元素值依次是( )。

A) 2    7    8    12    35                      B) 12    35    8    7    2  
C) 35    12    8    7    2                      D) 8    7    12    35    2

3、下面的程序段创建了 BufferedReader 类的对象 in, 以便读取本机 c 盘 my 文件夹下的文件 1.txt。File 构造函数中正确的路径和文件名的表示是( )。

```
File f = new File(填代码处);
file = new FileReader(f);
in = new BufferedReader(file);
```

A) "/1.txt"    B) "../my/1.txt"    C) "c:\\my\\1.txt"    D) "c:\\my\\1.txt"

4、应用程序的 main 方法中有以下语句, 则输出的结果是 ( )。

```
String s1 = new String("abc");
String s2 = new String("abc");
boolean b1 = s1.equals(s2);
boolean b2 = (s1 == s2);
System.out.print(b1 + "    " + b2);
```

A) true    false    B) false    true    C) true    true    D) false    false

5、应用程序的 main 方法中有以下语句, 则输出的结果是 ( )。

```
Hashtable hashtable = new Hashtable();
hashtable.put("x", "12345");
hashtable.put("y", "67890");
hashtable.put("a", "abcde");
System.out.println(hashtable.get("a"));
```

A) a    B) abcde    C) 12345    D) 67890

6、下面语句的功能是( )。

RandomAccessFile raf2 = new RandomAccessFile("1.txt","rw");

- A) 打开当前目录下的文件 1.txt, 既可以向文件写数据, 也可以从文件读数据。  
B) 打开当前目录下的文件 1.txt, 但只能向文件写入数据, 不能从文件读取数据。  
C) 打开当前目录下的文件 1.txt, 但不能向文件写入数据, 只能从文件读取数据。  
D) 以上说法都不对。

7、应用程序的 main 方法中有以下语句, 则输出的结果是 ( )。

```
String s="12345#aaa#bbb#67890";  
int n=s.indexOf("#");  
int k=s.indexOf("#",n+1);  
int m=s.indexOf("#",k+1);  
String s2=s.substring(m+1);  
System.out.println(s2);
```

- A) 123456      B) 67890      C) aaa      D) bbb

8、下面程序段执行后 b 的值是 ( )。

```
Integer integ = new Integer(9);  
boolean b = integ instanceof Object;
```

- A) 9      B) true      C) 1      D) false

9、应用程序 Test.java 的源程序如下, 在命令行键入:

```
java Test aaa bb c
```

回车后输出的结果是 ( )。

```
public class Test {  
    public static void main(String args[]) {  
        int k1=args.length;  
        int k2=args[1].length();  
        System.out.print(k1+" "+k2);  
    }  
}
```

- A) 3 2      B) 1 2      C) 1 3      D) 3 3

10、应用程序的 main 方法中有以下语句, 则输出的结果是 ( )。

```
int b[][]={{1},{2,2},{2,2,2}};  
int sum=0;  
for(int i=0;i<b.length;i++){  
    for(int j=0;j<b[i].length;j++){  
        sum*=b[i][j];
```

```

    }
}
System.out.println("sum="+sum);

```

A) 32      B) 11      C) 2      D) 3

11、应用程序的 main 方法中有以下语句，则执行后输出的结果是 ( )。

```

int[] x={125,21,5,168,98};
int max=x[0];
for(int i=1;i<x.length;i++){
    if(x[i]>max)
        max =x[i];
}
System.out.println(max);

```

A) 125      B) 5      C) 98      D) 168

12、程序 Test.java 编译运行后输出的结果是( )。

```

public class Test {
    String s1="java";
    public static void main(String args[]) {
        int z=2;
        Test t=new Test();
        System.out.println(t.s1+z);
    }
}

```

A) java2      B) 2      C) 没有输出结果      D) java

13、应用程序的 main 方法中有以下语句，则输出的结果是 ( )。

```

String s1="0.5", s2="12";
double x=Double.parseDouble(s1);
int y=Integer.parseInt(s2);
System.out.println(x+y);

```

A) 12.5      B) 120.5      C) 12      D) "12.5"

14、下面的程序创建了一个文件输出流对象，用来向文件 test.txt 中输出数据，假设程序当前目录下不存在文件 test.txt，编译下面的程序 Test.java 后，将该程序运行 3 次，则文件 test.txt 的内容是( )。

```

import java.io.*;
public class Test {

```

```

public static void main(String args[]) {
    try {
        String s="ABCDE";
        byte b[]=s.getBytes();
        FileOutputStream file=new FileOutputStream("test.txt",true);
        file.write(b);
        file.close();
    }
    catch(IOException e) {
        System.out.println(e.toString());
    }
}

```

A) ABCABC    B) ABCDE    C) Test    D) ABCDE ABCDE ABCDE

15、下面那些方法不是接口 Collection 中已声明的方法( )?

- A) 添加元素的 add(Object obj) 方法
- B) 删除元素的 remove(Object obj) 方法
- C) 得到元素个数的 length() 方法
- D) 返回迭代器的 iterator() 方法, 迭代器用于元素遍历

## 二、多选题

1、下面 main() 方法的定义哪些是正确的( )?

- A) public static void main(String args[]) {}    B) public static void main(String[]) {}
- C) public static void main(String[] args) {}    D) public static void main(String [] x) {}

2、线程对象的生命周期中,通常经历下面哪些状态( )。

- A) 新建    B) 运行    C) 就绪    D) 死亡

3、下面说法正确的是( )。

- A) final 可修饰类、属性(变量)、方法。
- B) abstract 可修饰类、方法。
- C) 抽象方法只有方法头, 没有方法体。
- D) 关键字 final 和 abstract 不能同时使用。



4、下面关于 java 中输入/输出流的说法正确的是( )。

- A) FileInputStream 与 FileOutputStream 类用读、写字节流。
- B) Reader 与 Writer 类用来读、写字符流。
- C) RandomAccessFile 既可以用来读文件，也可以用来写文件。
- D) File 类用来处理与文件相关的操作。

5、下面说法正确的是( )。

- A)如果某公共接口被声明在一个源文件中，则接口必须与源文件具有相同名字。
- B)如果源文件包含 import 语句，则该语句必须是除空行和注释行外的第一个语句行。
- C)如果某源文件包含 package 语句，则该语句必须是第一个非空、非注释行。
- D)接口中的方法必定是抽象方法，但可以不用关键字 abstract 修饰。

6、下面关于 Java 中异常处理 try 块的说法正确的是( )?

- A) try 块后通常应有一个 catch 块，用来处理 try 块中抛出的异常。
- B) catch 块后必须有 finally 块。
- C)可能抛出异常的方法调用应放在 try 块中。
- D)对抛出的异常的处理必须放在 try 块中

7、下面关于进程、线程的说法正确的是( )。

- A)进程是程序的一次动态执行过程。一个进程在其执行过程中，可以产生多个线程——多线程，形成多条执行线索。
- B)线程是比进程更小的执行单位，是在一个进程中独立的控制流，即程序内部的控制流。线程本身不能自动运行，栖身于某个进程之中，由进程启动执行。
- C)Java 多线程的运行与平台相关。
- D)对于单处理器系统，多个线程分时间片获取 CPU 或其他系统资源来运行。对于多处理器系统，线程可以分配到多个处理器中，从而真正的并发执行多任务。

8、关于 BufferedReader 类的 readLine()方法，以下说法正确的是 ( )。

- A)方法 readLine()每次读取一行数据。
- B)方法 readLine()每次读取一个字节。
- C)该方法可能抛出 IOException 异常，调用该方法时通常应将它放到 try 块中，并通过 catch 块处理异常。
- D)如果读到流的末尾，该方法返回的结果为 null。

### 三、程序阅读题

1、阅读下面的程序，回答问题(①问 3 分，②问 3 分，共 6 分)。

```
import java.awt.*;
import javax.swing.*;
public class T extends JFrame {
    public T () {
        super("GridLayout");
        Container con=this.getContentPane();
        con.setLayout(new GridLayout(2,3));
        con.add(new JButton("a"));
        con.add(new JButton("b"));
        con.add(new JButton("c"));
        con.add(new JButton("d"));
        con.add(new JButton("e"));
        con.add(new JButton("f"));
        setSize(200, 80);
        setVisible(true);
    }
    public static void main(String args[]) {
        new T();
    }
}
```

①画图表示程序运行后的图形界面。

②如果程序通过实现某个接口处理按钮的动作事件，则该接口名为何？接口中的方法头声明如何？

答案：



②接口名： ActionListener

接口中的方法： public void actionPerformed(ActionEvent e)

2、阅读下面的程序，回答问题(①问 3 分，②问 3 分，共 6 分)。

```
import java.util.*;
public class T {
```

```

public static void main(String args[]) {
    Set set = new TreeSet();
    set.add(new Integer(10));
    set.add(new Integer(5));
    set.add(new Integer(15));
    set.add(new Integer(5));
    set.add(new Integer(10));
    System.out.println("size = " + set.size());
    Iterator it=set.iterator();
    while(it.hasNext()){
        System.out.print(it.next()+" ");
    }
}

```

①程序运行后输出的结果如何？

②说明 java 中的集合(Set 接口)和映射(Map 接口)的主要区别。

答案:

①

size = 3

5 10 15

②Set 接口是不包含重复元素的集合；Map 接口将键映射到值，键可以重复，但每个键最多只能映射一个值。

3、阅读下面的程序，并回答问题(①问 3 分，②问 3 分，共 6 分)。

```

import java.io.*;

public class Test {

    public static void main(String args[]) throws IOException {

        BufferedReader buf=new BufferedReader(
            new InputStreamReader(System.in));

        while(true) {

            String str=buf.readLine();

            if(str.equals("quit"))
                break;

            int x=Integer.parseInt(str);

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

```

```

    }
}
}

```

编译运行上面的程序：

- ❶从键盘输入 5，回车后输出的结果如何？
- ❷从键盘输入 quit，回车后程序执行情况如何？

答案：

- ❶25
- ❷终止应用程序的运行。

## 《JAVA 程序设计》期末考试试题 （五）

### 二、单选题

1、当某一线程正处于休眠状态，而另一个线程用 Thread 类中的 interrupt() 方法中断它时，抛出的异常类型是（ ）。

- A) IOException
- B) RuntimeException
- C) InterruptedException
- D) ClassNotFoundException

2、下面的程序段的功能是( )。

```

File file1=new File("d:\\xxx\\yyy\\zzz");
file1.mkdirs();

```

- A)在当前目录下生成子目录： \xxx\yyy\zzz
- B)生成目录： e:\xxx\yyy\zzz
- C)在当前目录下生成文件 xxx.yyy.zzz
- D)以上说法都不对

3、应用程序的 main 方法中有以下语句，则输出的结果是（ ）。

```

String s = "xxxxxxxxxxxxxx#123#456#zzzzz";
int n = s.indexOf("#");
int k = s.indexOf("#", n+1);
String s2 = s.substring(n+1, k);
System.out.println(s2);

```

A) 123456      B) 123      C) xxxxxxxxxxxxxxxx      D) zzzzz

4、关于下面的程序 Test.java 说法正确的是( )。

```
public class Test {  
    String x="1";  
    int y;  
    public static void main(String args[]) {  
        int z=2;  
        System.out.println(x+y+z);  
    }  
}
```

A)3      B)102      C) 12      D)程序有编译错误

5、应用程序的 main 方法中有以下语句，则输出的结果是 ( )。

```
int b[][]={{1, 1, 1}, {2,2}, {3}};  
int sum=0;  
for(int i=0; i<b.length; i++) {  
    for(int j=0; j<b[i].length; j++) {  
        sum+=b[i][j];  
    }  
}  
System.out.println("sum="+sum);
```

A) 10      B)6      C) 9      D) 13

6、应用程序的 main 方法中有以下语句，则执行后输出的结果是 ( )。

```
int[] x = {125,21,5,168,98};  
int min = x[0];  
for (int i=1; i<x.length; i++){  
    if(x[i] < min)  
        min = x[i];  
}  
System.out.println(min);
```

A) 125      B) 5      C) 98      D) 168

7、应用程序的 main 方法中有以下语句，则输出的结果是 ( )。

```
Hashtable hashtable=new Hashtable();
hashtable.put("100","aaa");
hashtable.put("200","bbb");
hashtable.put("300","ccc");
System.out.println(hashtable.get("300").toString()
                    + hashtable.get("200").toString()
                    + hashtable.get("100").toString());
```

A) aaa    B) bbb    C) ccc    D) cccbbbaaa

8、以下关于 java 异常说法不正确的是 ( )。

- A) Throwable 类是 Java 语言中 Error 类和 Exception 类的父类。
- B) 当异常对象是 Exception 类 (或其子类) 的实例时，能通过 Java 虚拟机或者 throw 语句抛出该异常对象，并能通过 try...catch...finally 处理。
- C) 如果只用一个 catch 块捕捉多个异常对象，则 catch 子句中的参数类型应是所有异常对象的父类。
- D) 以上说法都不对。

9、下面的程序创建了一个文件输出流对象，用来向文件 test.txt 中输出数据，假设程序当前目录下不存在文件 test.txt，编译下面的程序 Test.java 后，将该程序运行 3 次，则文件 test.txt 的内容是 ( )。

```
import java.io.*;

public class Test {

    public static void main(String args[]) {

        try {

            String s="ABCDE";
            byte b[]=s.getBytes();
            FileOutputStream file=new FileOutputStream("test.txt", true);
            file.write(b);
            file.close();

        }

        catch(IOException e) {

            System.out.println(e.toString());

        }

    }

}
```

```
    }  
}
```

A) ABCABC    B) ABCDE    C) Test    D) ABCDE ABCDE ABCDE

10、下面说法不正确的是( )?

- A) 列表(List)、集合(Set)和映射(Map)都是 java.util 包中的接口。
- B) List 接口是可以包含重复元素的有序集合。
- C) Set 接口是不包含重复元素的集合。
- D) Map 接口将键映射到值，键可以重复，但每个键最多只能映射一个值。

## 二、多选题

1、下面哪些关键字能用来控制对类成员的访问( )?

- A) public    B) protected    C) private    D) default

2、下面关于 Java 中异常处理 try 块的说法正确的是( )?

- A) try 块后通常应有一个 catch 块，用来处理 try 块中抛出的异常。
- B) catch 块后必须有 finally 块。
- C) 可能抛出异常的方法调用应放在 try 块中。
- D) 对抛出的异常的处理必须放在 try 块中

3、线程对象的生命周期中,通常经历下面哪些状态 ( )。

- A) 新建    B) 运行    C) 就绪    D) 死亡

4、定义如下的二维数组 b，下面的说法正确的是( )。

```
int b[][]={{1, 2, 3}, {4, 5}, {6, 7, 8}};
```

- A) b.length 的值是 3。    B) b[1].length 的值是 3。
- C) b[1][1] 的值是 5。    D) 二维数组 b 的第一行有 3 个元素

5、类 B 是一个抽象类，类 C 是类 B 的非抽象子类，下列创建对象 x1 的语句中正确的是( )。

- A) B x1=new B();    B) B x1=new C();
- C) C x1=new C();    D) C x1=new B();

6、下面关于进程、线程的说法正确的是( )。

A)进程是程序的一次动态执行过程。一个进程在其执行过程中，可以产生多个线程——多线程，形成多条执行线索。

B)线程是比进程更小的执行单位，是在一个进程中独立的控制流，即程序内部的控制流。线程本身不能自动运行，栖身于某个进程之中，由进程启动执行。

C)Java 多线程的运行与平台相关。

D)对于单处理器系统，多个线程分时间片获取 CPU 或其他系统资源来运行。对于多处理器系统，线程可以分配到多个处理器中，从而真正的并发执行多任务。

7、下面关于外部类和内部类成员的相互访问的规则中，正确的选项是( )。

A)内部类可以直接访问外部类的成员

B)外部类可以直接访问内部类的成员

C)外部类不能访问内部类的成员

D)外部类可通过创建内部类的实例来访问内部类的成员

### 三、程序阅读题

1、阅读下面的程序代码，并回答问题(①问 3 分，②问 3 分，共 6 分)。

```
String s1 = new String("abcde");
String s2 = new String("abcde");
boolean b1 = s1.equals(s2);
boolean b2 = s1 == s2;
System.out.print(b1+" "+b2);
```

①程序段执行后，在命令行的输出结果如何？

②解释输出(1)的结果的原因？

答案：

① true false

②equals 方法比较两个字符串的内容是否相等；运算符“==”判断两个对象是否指向同一个引用，即是否为同一个对象。

2、阅读下面的程序，并回答问题(①问 3 分，②问 3 分，共 6 分)。

```
import java.io.*;

public class Test {

    public static void main(String args[]) throws IOException {

        BufferedReader buf=new BufferedReader(

            new InputStreamReader(System.in));

        while(true) {
```



}  
}  
A) ABCABC    B) ABCDE    C) Test    D) ABCDE ABCDE ABCDE

10、下面说法不正确的是( )?

- A) 列表(List)、集合(Set)和映射(Map)都是 java.util 包中的接口。
- B) List 接口是可以包含重复元素的有序集合。
- C) Set 接口是不包含重复元素的集合。
- D) Map 接口将键映射到值，键可以重复，但每个键最多只能映射一个值。

## 二、多选题

1、下面哪些关键字能用来控制对类成员的访问( )?

- A) public    B) protected    C) private    D) default

2、下面关于 Java 中异常处理 try 块的说法正确的是( )?

- A) try 块后通常应有一个 catch 块，用来处理 try 块中抛出的异常。
- B) catch 块后必须有 finally 块。
- C) 可能抛出异常的方法调用应放在 try 块中。
- D) 对抛出的异常的处理必须放在 try 块中

3、线程对象的生命周期中,通常经历下面哪些状态( )。

- A) 新建    B) 运行    C) 就绪    D) 死亡

4、定义如下的二维数组 b，下面的说法正确的是( )。

```
int b[][]={{1, 2, 3}, {4, 5}, {6, 7, 8}};
```

- A) b.length 的值是 3。    B) b[1].length 的值是 3。
- C) b[1][1] 的值是 5。    D) 二维数组 b 的第一行有 3 个元素

5、类 B 是一个抽象类，类 C 是类 B 的非抽象子类，下列创建对象 x1 的语句中正确的是( )。

- A) B x1=new B();    B) B x1=new C();
- C) C x1=new C();    D) C x1=new B();

6、下面关于进程、线程的说法正确的是( )。

A)进程是程序的一次动态执行过程。一个进程在其执行过程中,可以产生多个线程——多线程,形成多条执行线索。

B)线程是比进程更小的执行单位,是在一个进程中独立的控制流,即程序内部的控制流。线程本身不能自动运行,栖身于某个进程之中,由进程启动执行。

C)Java 多线程的运行与平台相关。

D)对于单处理器系统,多个线程分时间片获取 CPU 或其他系统资源来运行。对于多处理器系统,线程可以分配到多个处理器中,从而真正的并发执行多任务。

7、下面关于外部类和内部类成员的相互访问的规则中,正确的选项是( )。

A)内部类可以直接访问外部类的成员

B)外部类可以直接访问内部类的成员

C)外部类不能访问内部类的成员

D)外部类可通过创建内部类的实例来访问内部类的成员

### 三、程序阅读题

1、阅读下面的程序代码,并回答问题(①问 3 分, ②问 3 分, 共 6 分)。

```
String s1 = new String("abcde");
String s2 = new String("abcde");
boolean b1= s1.equals(s2);
boolean b2 = s1== s2;
System.out.print(b1+" "+b2);
```

①程序段执行后,在命令行的输出结果如何?

②解释输出(1)的结果的原因?

答案:

① true false

②equals 方法比较两个字符串的内容是否相等;运算符“==”判断两个对象是否指向同一个引用,即是否为同一个对象。

2、阅读下面的程序,并回答问题(①问 3 分, ②问 3 分, 共 6 分)。

```
import java.io.*;

public class Test {

    public static void main(String args[]) throws IOException {

        BufferedReader buf=new BufferedReader(
            new InputStreamReader(System.in));

        while(true) {
```

```

        String str = buf.readLine();
        if(str.equals("quit"))
            break;
        int x=Integer.parseInt(str);
        System.out.println(x*x);
    }
}
}

```

编译运行上面的程序：

- ①从键盘输入 10，回车后输出的结果如何？
- ②从键盘输入 exit，回车后程序能正确执行吗？为什么？

答案：

①100

②不能；因为方法 Integer.parseInt(str)不能将字符串“exit”转化为整数，抛出异常。

3、写出下面的程序编译、运行后的结果(6分)。

```

public class Test{
    public static void main(String args[]) {
        new Student("Tom", 'm', 90, 88);
        new Student("Jack", 'm', 66, 89);
        new Student("Mary", 'f', 76, 86);
        System.out.println("name\tsex\tchinese\tenglish");
        Student.print();
    }
}

```

```

class Student {
    protected String name;
    protected char sex;
    protected int chinese;
    protected int english;
    protected Student next;
    static Student list;

    Student (String name, char sex, int chinese, int english) {

```

```

        this.name=name;
        this.sex=sex;
        this.chinese=chinese;
        this.english=english;
        this.next=list;
        list=this;
    }
    static void print() {
        Student friend=list;
        if (friend==null)
            System.out.println("The list is empty.");
        else {
            do{
                System.out.println(friend.toString());
                friend=friend.next;
            }while(friend!=null);
        }
    }
    public String toString() {
        return new String(name+"\t"+sex+"\t"+chinese+"\t"+english);
    }
}

```

答案:

name	sex	chinese	english
Mary	f	76	86
Jack	m	66	89
Tom	m	90	88

## 《JAVA 程序设计》期末考试试题 (六)

### 一、填空题

1. 定义类的保留字是( `class` ), 定义接口的保留字是( `interface` )。
2. Socket 通常也称为 ( 套接字 ), 用于描述( IP 地址 ) 和 ( 端口 )。
3. 线程的优先级在( 1 )至( 10 )之间, 数值越大( 任务越紧急 )。
4. 构造方法是一种特殊的成员方法, 构造方法名与( 类名 )相同。
5. Java 语言只允许单继承, 指每个类只能有一个 ( 父类 )。
6. Java 源程序的扩展名是( `.java` ), 经过编译后的程序的扩展名是 ( `.class` )。
7. 在一个时间只能由一个线程访问的资源称为 ( 临界资源 )。访问临界资源的代码( 临界代码 )。
8. 在多线程系统中, 多个线程之间有 ( 同步 ) 和 ( 互斥 ) 两种关系。

## 二、选择题

1. 关于选择结构下列哪个说法正确? ( )
  - A. if 语句和 else 语句必须成对出现
  - B. if 语句可以没有 else 语句对应
  - C. switch 结构中每个 case 语句中必须用 break 语句
  - D. switch 结构中必须有 default 语句
2. while 循环和 do...while 循环的区别是: ( )
  - A. 没有区别, 这两个结构任何情况下效果一样
  - B. while 循环比 do...while 循环执行效率高
  - C. while 循环是先循环后判断, 所以循环体至少被执行一次
  - D. do...while 循环是先循环后判断, 所以循环体至少被执行一次
3. 关于 for 循环和 while 循环的说法哪个正确? ( )
  - A. while 循环先判断后执行, for 循环先执行后判断。
  - B. while 循环判断条件一般是程序结果, for 循环的判断条件一般是非程序结果
  - C. 两种循环任何时候都不可以替换
  - D. 两种循环结构中都必须有循环体, 循环体不能为空
4. 下列修饰符中与访问控制无关的是 ( )
  - A. private
  - B. public
  - C. protected
  - D. final

5. void 的含义: ( )

- A. 方法没有返回值
- B. 方法体为空
- C. 没有意义
- D. 定义方法时必须使用

6. return 语句: ( )

- A. 只能让方法返回数值
- B. 方法都必须含有
- C. 方法中可以有多句 return
- D. 不能用来返回对象

7. 关于对象成员占用内存的说法哪个正确? ( )

- A. 同一个类的对象共用同一段内存
- B. 同一个类的对象使用不同的内存段, 但静态成员共享相同的内存空间
- C. 对象的方法不占用内存
- D. 以上都不对

8. 下列说法哪个正确?

- A. 不需要定义类, 就能创建对象
- B. 对象中必须有属性和方法
- C. 属性可以是简单变量, 也可以是一个对象
- D. 属性必须是简单变量

9. 下列说法哪个正确? ( )

- A. 一个程序可以包含多个源文件
- B. 一个源文件中只能有一个类
- C. 一个源文件中可以有多个公共类
- D. 一个源文件只能供一个程序使用

10. 关于方法 main () 的说法哪个正确? ( )

- A. 方法 main () 只能放在公共类中
- B. main() 的头定义可以根据情况任意更改
- C. 一个类中可以没有 main() 方法
- D. 所有对象的创建都必须放在 main() 方法中

11. 构造函数何时被调用? ( )

- A、创建对象时  
B、类定义时  
C、使用对象的方法时  
D、使用对象的属性时

12. 抽象方法: ( )

- A、可以有方法体  
B、可以出现在非抽象类中  
C、是没有方法体的方法  
D、抽象类中的方法都是抽象方法

13. 关于继承的说法正确的是: ( )

- A、子类将继承父类所有的属性和方法。  
B、子类将继承父类的非私有属性和方法。  
C、子类只继承父类 public 方法和属性  
D、子类只继承父类的方法，而不继承属性

14. 关于构造函数的说法哪个正确? ( )

- A、一个类只能有一个构造函数  
B、一个类可以有多个不同名的构造函数  
C、构造函数与类同名  
D、构造函数必须自己定义，不能使用父类的构造函数

15. this 和 super:

- A、都可以用在 main()方法中  
B、都是指一个内存地址  
C、不能用在 main()方法中  
D、意义相同

16. 关于 super 的说法正确的是:

- A、是指当前对象的内存地址  
B、是指当前对象的父类对象的内存地址  
C、是指当前对象的父类  
D、可以用在 main()方法中

17. 覆盖与重载的关系是 ( )

- A. 覆盖只有发生在父类与子类之间，而重载可以发生在同一个类中
- B. 覆盖方法可以不同名，而重载方法必须同名
- C. final 修饰的方法可以被覆盖，但不能被重载
- D. 覆盖与重载是同一回事

18. 关于接口哪个正确？ ( )

- A. 实现一个接口必须实现接口的所有方法
- B. 一个类只能实现一个接口
- C. 接口间不能有继承关系
- D. 接口和抽象类是同一回事

19. 异常包含下列哪些内容？ ( )

- A. 程序执行过程中遇到的事先没有预料到的情况
- B. 程序中的语法错误
- C. 程序的编译错误
- D. 以上都是

20. 对于已经被定义过可能抛出异常的语句，在编程时： ( )

- A. 必须使用 try / catch 语句处理异常，或用 throws 将其抛出
- B. 如果程序错误，必须使用 try / catch 语句处理异常
- C. 可以置之不理
- D. 只能使用 try / catch 语句处理

21. 字符流与字节流的区别在于 ( )

- A. 前者带有缓冲，后者没有
- B. 前者是块读写，后者是字节读写
- C. 二者没有区别，可以互换使用
- D. 每次读写的字节数不同

22. 下列流中哪个不属于字节流 ( )

- A. FileInputStream
- B. BufferedInputStream
- C. FilterInputStream
- D. InputStreamReader



### 三、程序填空题

```
1. public class Sum{
    public static void main(String [] args){
        int j=10;
        System.out.println("j is : "+j);
        calculate(j);
        System.out.println("At last, j is : "+j);
    }
    static void calculate (int j){
        for (int i = 0;i<10;i++)
            j++;
        System.out.println("j in calculate() is: "+j);
    }
}
```

输出结果为：

j is : (1)

j in calculate() is : (2)

At last j is : (3)

答案： (1) 10; (2) 20; (3) 10。

#### 2. 按要求填空

```
abstract class SuperAbstract{
    void a(){...}
    abstract void b();
    abstract int c(int i);
}
interface AsSuper
{
    void x();
}
abstract class SubAbstract extends SuperAbstract implements AsSuper
{
    public void b(){...}
    abstract String f();
}
```

```

}
public class InheritAbstract extends SubAbstract{
    public void x(){...}
    public int c(int i) {...}
    public String f(){...}
    public static void main(String args[]){
        InheritAbstract instance=new InheritAbstract();
        instance.x();
        instance.a();
        instance.b();
        instance.c(100);
        System.out.println(instance.f());
    }
}

```

在以上这段程序中：

抽象类有： SuperAbstract 和 (1) (写出类名)  
 非抽象类有： (2) (写出类名)  
 接口有： (3) (写出接口名)

AsSuper 中的 x()方法是 (4) 方法，所以在 InheritAbstract 中必须对它进行 (5)

答案：

(1) SuperAbstract;  
 (2) InheritAbstract;  
 (3) AsSuper;  
 (4) 抽象;  
 (5) 覆盖和实现。

### 3. 按注释完成程序

```

public class Leaf {
    private int i = 0; //此属性值用于检验
    Leaf increment(){ //定义方法 increment(), 返回值是 Leaf 类的对象
        i++;
        return (1) ;//将当前对象的地址作为返回值返回
    }
    void print() {

```

```

        System.out.println(" i = " + i);
    }
    public static void main(String args[]){
        Leaf x = (2); //创建 Leaf 类的对象 x
        x.increment().increment().increment().print();
    } //多次调用方法 increment(), 返回的都是 x 的地址, i 值表示调用次数
}

```

输出结果为 i = (3)

答案:

(1) this;

(2) new Leaf();

(3) 3

#### 4. 按注释提示完成文件复制的程序

//FileStream 源代码如下:

```

import java.io.*;
class FileStream {
    public static void main(String args []) {
        try {
            File inFile = new File("file1.txt"); //指定源文件
            File outFile = new File("file2.txt"); //指定目标文件
            FileInputStream fis = (1) ;
            FileOutputStream fos = new FileOutputStream(outFile);
            int c;
            //逐字节从源文件中输入, 再输出到 fos 流
            while ((c = fis.read ()) != -1)
                (2) ;
            fis.close();
            fos.close();
        }
        catch (Exception e) {
            System.out.println("FileStreamsTest: "+e);
        }
    }
}

```

```
}
```

答案:

(1) `new FileInputStream(inFile);`

(2) `fos.write(c);`

5. 阅读程序, 给出结果:

// AbstractClassDemo.java 源代码如下:

```
abstract class Shape {    //定义抽象类 Shape 和抽象方法 display
    abstract void display();
}

class Circle extends Shape {
    void display() {        //实现抽象类的方法
        System.out.println("Circle");
    }
}

class Rectangle extends Shape {
    void display() {        //实现抽象类的方法
        System.out.println("Rectangle");
    }
}

class Triangle extends Shape {
    void display() {        //实现抽象类的方法
        System.out.println("Triangle");
    }
}

public class AbstractClassDemo{
    public static void main(String args[]){
        (new Circle()).display(); //定义无名对象来调用对应的 display 方法
        (new Rectangle()).display();
        (new Triangle()).display();
    }
}
```

输出结果是 ?

答案: (1) Circle; (2) Rectangle; (3) Triangle。

## 《JAVA 程序设计》期末考试试题 （七）

### 一、选择题

1. 请说出下列代码的执行结果：

```
String s = "abcd";  
String s1 = new String(s);  
if (s == s1) System.out.println("the same");  
if (s.equals(s1)) System.out.println("equals");
```

- A. the same      equals      B. equals  
C. the same      D. 什么结果都不输出

2. 下列有关 Java 中接口的说法哪个是正确的？

- A. 接口中含有具体方法的实现代码  
B. 若一个类要实现一个接口，则用到 “implements” 关键字  
C. 若一个类要实现一个接口，则用到 “extends” 关键字  
D. 接口不允许继承

3. 下列代码的执行结果是什么？

```
String s1 = "aaa";  
s1.concat("bbb");  
System.out.println(s1);
```

- A. The string "aaa".  
B. The string "aaabbb".  
C. The string "bbbbaaa".  
D. The string "bbb".

4. 如果有一个对象 myListener ( 其中 myListener 对象实现了 ActionListener 接口 ), 下列哪条语句使得 myListener 对象能够接受处理来自于 smallButton 按钮对象的动作事件？

- A. smallButton.add(myListener);  
B. smallButton.addListener(myListener);  
C. smallButton.addActionListener(myListener);  
D. smallButton.addItem(myListener);

### 二、读程序题

1. 读下列代码，说出这段程序的功能。

```
import java.io.*;
public class Test{
    public static void main( String [] argv) {
        try {
            BufferedReader is =
                new BufferedReader( new InputStreamReader(System.in));
            String inputLine;
            While ((inputLine = is.readLine ())!= null) {
                System.out.println(inputLine);
            }
            is.close();
        }catch (IOException e) {
            System.out.println("IOException: " + e);
        }
    }
}
```

答案：读取键盘输入，显示到屏幕上，直到键盘输入为空为止。

2、 读下列程序，写出正确的运行结果。

```
class test {
    public static void main (String [] args ){
        int x=9, y;
        if (x>=0)
            if (x>0)
                y=1;
            else y=0;
        else y=-1;
        System.out.println(y);
    }
}
```

答案： 1

3、 读程序，写出正确的运行结果。

```

public class Father{
    int a=100;
    public void miner(){
        a--;
    }
    public static void main(String[] args){
        Father x = new Father();
        Son y = new Son();
        System.out.println(y.a);
        System.out.println(y.getA());
        y.miner();
        System.out.println(y.a);
        System.out.println(y.getA());
    }
}
class Son extends Father{
    int a = 0;
    public void plus(){
        a++;
    }
    public int getA() {
        return super.a;
    }
}

```

答案:

0

100

0

99

### 三. 简答题

1. Java 语言的特点。

答:

简单性: Java 风格类似于 C++, 但它摒弃了 C++ 中容易引起程序错误的地方

面向对象：Java 语言的设计是完全面向对象

分布式：

解释执行：

健壮性：Java 提供自动垃圾回收机制，异常处理机制，进行严格的类型检查

平台无关性：

安全性

多线程

动态性

2. 请描述 AWT 事件模型。

答：

结合 AWT 事件模型并举例来说：

```
import java.awt.event.*;
```

1. 监听对象必须实现对应事件监听器的接口

```
class MyFirstFrame extends Frame implements ActionListener  
{...}
```

2. 明确事件监听器的接口形式

```
public void actionPerformed ( ActionEvent event) {...}
```

3. MyFirstFrame 类必须实现接口 ActionListener 中的所有方法。

4. 注册监听对象。

为了把 MyFirstFrame 对象注册为两个按钮的事件监听对象，必须在 MyFirstFrame 的构造函数中添加语句如下：

```
cancelButton.addActionListener(this);
```

```
okButton.addActionListener(this);
```

3. 在 Java 中，怎样创建一个线程？

答：

1、定义类来实现 Runnable 接口

```
class TestThread implements Runnable {  
    public void run() { ... }  
}
```

2、继承 Thread 类来实现

```
class TestThread extends Thread {  
    TestThread(String name) {  
        super(name);
```



```
        start();  
    }  
    public void run() {...}  
}
```



## 《JAVA 面向对象程序设计》试题与解析库

1) 在 JAVA 编程中, 关于 Font 下列 ( ) 是正确的。(选择两项)

- a) 在我们的程序中可以使用 Font 类中定义的字体常量。
- b) 我们可以使用 Toolkit 类中的函数来获取我们的机器中 java 所支持的字体的列表
- c) 我们可以自己创建字体实例对象
- d) 以上说明都正确

【解析】参考答案: AB

Font 属于 java.awt 包中的一个类, 专门用来设置程序中的字体实例对象。

创建 Font 类的对象时使用的方法: `getFont(int face,int style,int size);`

例如: `Font font = Font.getFont(Font.FACE_SYSTEM,Font.STYLE_BOLD,Font.SIZE_MEDIUM);`

无论哪一个参数, 都只能使用系统设置的数值, 这些数值具体的大小在不同的手机上可能不同。下面对于其中的三个参数的取值做详细的介绍:

face 参数指字体的外观; style 参数指字体的样式; size 参数指字体的大小。这三个参数都是系统中定义好的一些常量。

获得系统的默认字体: `Font font = Font.getDefaultFont();`

Toolkit 类是 Abstract Window Toolkit 的所有实际实现的抽象父类。Toolkit 用于把各种组件绑定到特定的本地工具箱实现上。

该类有个方法 `getFontList()`

用于返回该工具箱中可用字体名。也就是可以获取机器中 java 所支持的字体的列表。

2) 在 Java 语言中, 在包 Package1 中包含包 Package2, 类 Class\_A 直接隶属于包 Package1, 类 Class\_B 直接隶属于包 Package2。在类 Class\_C 要应用 Class\_A 的方法 A 和 Class\_B 的方法 B, 你需要 ( ) 语句。(选择两项)

- a) `import Package1.*;`
- b) `import Package1.Package2.*;`
- c) `import Package2.*`
- d) `import Package2.Package1.*;`

【解析】AB

Java 语言中的包机制就像 Windows 中的文件夹, 用来归类类文件的位置。要使用包中的类, 有如下二种方法:

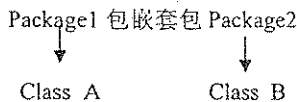
直接导入该包中要用到的类。例如要用到 Color 类, 可以

`import java.awt.Color`

干脆将包中所有的类直接导入。但此时如果包有嵌套的包, 不能将嵌套包中的类倒入。

`Import java.awt.*`

本题包的关系如下:



掌握了这两条, 要在类 Class\_C 要应用 Class\_A 的方法 A 和 Class\_B 的方法 B。也就是要在 Class\_C 中导入 Class\_A 和 Class\_B。方法如下:

`import Package1.* //可以导入 Class_A`

`import Package1.Package2.* //可以导入 Class_B`

```

3) import java.awt.*;
   import java.applet.*;
   public class ButtonDemo extends Applet {
       Public void init()
       {
           Button pushBotton=new Button(“ok”);
           Button downButton=new Button(“Yes”);
           add(pushBotton);
           add(downBotton);}
       }

```

根据以上代码，下列解释正确的是（）。（选择两项）

- a) 该代码画了一个按钮
- b) Button(“ok”)创建一个有显示“ok”的按钮
- c) Button()是构造函数
- d) 按钮属于容器

【解析】BC

Button 是一个 GUI 组件，并不是容器。因为它里面不能再放其他组件。要创建按钮，可以使用 Button 类创建。

```
Button pushBotton=new Button(“ok”)
```

其中 pushBotton 是按钮实例对象名，Button() 是构造方法，初始化该按钮，此处带有参数 ok，表示该按钮所显示的文本为 OK。等价于 setText 方法的作用。此代码中的 add 方法是将按钮添加到容器。但此时该按钮并没有显示出来，因为还没有实现画按钮。

4) 在 JAVA 编程中，关于 Graphics，下面（）是正确的。（选择一项）

- a) 在这个类中定义了一些基本的绘图方法
- b) 这个类还存在一些不足，因此出现了 Graphics2D 类，弥补了这个类的某些不足
- c) 这个类是一个抽象类，我们不能创建这个类的实例
- d) 以上说法都正确

【解析】D

类 java.awt.Graphics，

定义：public abstract class Graphics extends Object

Graphics 类是所有图形上下文的抽象基类，该类包含了一些基本的绘图方法。这个上下文允许应用将图形绘制到由不同设备实现的组件上，以及绘制到空闲屏幕的映像中。

由于 Graphics 是一个抽象类，应用不能直接调用该构造函数。图形上下文是从其他图形上下文获得的或是通过在一个组件上调用 getGraphics 创建的。例如使用如下这二个方法：

create, getGraphics

针对该类的一些缺点，出现了 Graphics2D 类，利用 java.awt.Graphics2D 类可以绘制各种图形，矩形，圆，二次曲线，饼形以及它们的填充图形等。Java 2D 图形编程/2D Graphic。

5) 在 JAVA 编程中，将鼠标放在按钮上以后，用鼠标单击按钮，将会发生鼠标事件和组件激活事件，就鼠标事件而言，将调用（）个监听器方法。（选择一项）

- a) 1
- b) 2
- c) 3
- d) 4

**【解析】B**

鼠标事件在单击按钮时实际包含了二个事件：

鼠标压下 和 鼠标 弹起

在 Java 的事件机制中，每个事件都必须有一个事件监听者。故选 B

6) JAVA 中，为了辨别用户关闭窗口的时间，要实现监听器接口（）。（选择一项）

- a) MouseListener
- b) ActionListener
- c) WindowListener
- d) 以上都要

**【解析】D**

在 Java 程序中这一般是通过 实现适当的事件监听者接口来完成的。比如如果需要响应按钮事件，就需要实现 ActionListener 监听者接口；如果需要响应窗口事件，就需要实现 WindowListener 监听者接口。此处当用鼠标点击关闭时，会产生一个 MouseEvent，此类需要实现 MouseListener 接口。故本题选 D

7) 在 Java 语言中，如果你有下面的类定义：

```
abstract class Shape {  
    abstract void draw();  
}
```

```
class Square extends Shape{}
```

如果你试图编译上面的代码会发生（）。（选择一项）

- a) 一切成功编译
- b) Shape 可以编译 Square 不能编译
- c) Square 可以编译 Shape 不能编译
- d) Shape, Square 都不能编译

**【解析】D**

由于 Square 类没有实现父类 Shape 的方法 draw，从而导致这二个类都不能编译。

8) int[] my\_Array;

```
my_Array=new int[5];
```

```
for(int count=0;count<=5;count++)
```

```
    System.out.println(my_Array[count]);
```

以上 Java 代码运行的结果是（）。（选择一项）

- a) 将 1, 2, 3, 4, 5 输出到屏幕
- b) 将 0, 1, 2, 3, 4 输出到屏幕
- c) 将 0, 1, 2, 3, 4, 5 输出到屏幕
- d) 将出现运行时异常

**【解析】D**

数组 my\_Array 包含 5 个元素，元素的索引从 0 到 4；而在 for 循环体中出现了当 count=5 时，会超出索引边界；故 my\_Array[5] 会产生异常。

9) 下面描述（）是正确的。（选择两项）

- a) Java 的源程序必须以 ".java" 或 ".jav" 扩展名保存
- b) Java VM 可以是软件也可以是硬件

- c) Java 使用解释器执行代码
- d) 用高级语言编写的代码可以让计算机理解并执行它们

【解析】B C

Java 的源程序必须以".java"结尾。

高级语言编写的程序必须经过编译或解释为低级语言;也即要翻译为计算机可以理解的机器语言。

Java VM (虚拟机) 可以是软件也可以是硬件。

Java 使用解释器执行代码。

10) 在 JAVA 编程中, 基于线程的多任务处理环境中, 执行特定任务的可执行代码的最小单位是 ( )。(选择一项)

- a) 进程
- b) 线程
- c) 应用程序
- d) 服务

【解析】B

进程是由一个或多个线程组成, 进程中可执行代码的最小单位就是线程。

11) 下列选项中, 属于 Java 语言的关键字的是 ( )。(选择两项)

- a) goto
- b) malloc
- c) extends
- d) FALSE

【解析】A C

先做下面几个题目:

1, which of the following are keywords or reserved words in java ?

a) if b)then c)goto d)while e)case f)sizeof

2, which of the following are java key words ?

a)double b)Switch c)then d)instanceof

3, which of these are key words in java ?

a) default b)NULL c)String d)throws e)long f>true

答案: 1, acde 2, ad 3, adef

解释来了:

1, then 和 sizeof 都不是 java 的关键字, 熟悉 c 或者 c++, 写惯了 asp 的高手就要小心喽。

2, 所有的关键字都是小写的, 所以 Switch 不是关键字。instanceof 看上去像方法名, 但其实是关键字;

3, 大写的 NULL 不是 java 语言的关键字。String 是 java 语言的一个封装类的类名, 也不是关键字。

再来点系统的:

正确识别 java 语言的关键字 (keyword) 和保留字 (reserved word) 是十分重要的。Java 的关键字对 java 的编译器有特殊的意义, 他们用来表示一种数据类型, 或者表示程序的结构等。保留字是为 java 预留的关键字, 他们虽然现在没有作为关键字, 但在以后的升级版本中有可能作为关键字。

关键字列表

abstract boolean break byte case  
catch char class continue default  
do double else extends false  
final finally float for if  
implements import instanceof int interface  
long native new null package  
private protected public return short  
static super switch synchronized this  
throw throws transient true try  
void volatile while  
保留字  
const, goto

注意:

识别 java 语言的关键字, 不要和其他语言如 c/c++ 的关键字混淆。  
const 和 goto 是 java 的保留字。  
所有关键字都是小写  
friendly, sizeof 不是 java 的关键字

12) 在 JAVA 语言中, 使我们能够使用和更改字体来显示或键入文本的类是 ( )。(选择一项)

- a) Java.awt.Font
- b) Java.awt.Graphics.Font
- c) Java.Graphics.Font
- d) Java.Font

【解析】A

Font 类位于 java.awt 包中。

13) 在 Java 中, 关键字 ( ) 使类不能派生出子类。(选择一项)

- a) final
- b) public
- c) private
- d) volatile
- e) native

【解析】A

在类的前面加了 final 关键字, 说明该类是一个最终的类, 当然也就不能被别人继承, 也就不能派生出子类了。

14) 在 JAVA 编程中, Swing 包中的组件处理事件时, 下面 ( ) 是正确的。(选择一项)

- a) Swing 包中的组件也是采用事件的授权处理模型来处理事件的
- b) Swing 包中的组件产生的事件类型, 也都带有一个 J 字母, 如: JMouseEvent
- c) Swing 包中的组件也可以采用事件的传递处理机制
- d) Swing 包中的组件所对应的事件适配器也是带有 J 字母的, 如: JMouseAdapter

【解析】A

Swing 是由 100% 纯 Java 实现的, Swing 组件是用 Java 实现的轻量级 (light-weight) 组件, 没有本地代码, 不依赖操作系统的支持, 这是它与 AWT 组件的最大区别。由于 AWT 组件

通过与具体平台相关的对等类 (Peer) 实现, 因此 Swing 比 AWT 组件具有更强的实用性。Swing 在不同的平台上表现一致, 并且有能力提供本地窗口系统不支持的其它特性。Swing 包中的组件也是采用事件的授权处理模型来处理事件的。它的事件类型和事件适配器类依然是按照 awt 包中的方式来定义的, 并没有附带一个 J 字母。

- 15) 在 JAVA 编程中, Java 编译器会将 java 程序转换为 ( )。(选择一项)
- a) 字节码
  - b) 可执行代码
  - c) 机器代码
  - d) 以上所有选项都不正确

【解析】A

Java 编译器将 java 程序转换为字节码。

- 16) 在 JAVA 编程中, 以下 ( ) 命令用来执行 java 类文件。(选择一项)
- a) javac
  - b) java
  - c) appletviewer
  - d) 以上所有选项都不正确

【解析】B

Javac 是用来编译 java 的源文件的命令。

Java 执行经过编译后产生的类文件。

Appletviewer 是用来查看 java Applet 程序的工具。

- 17) 在 Java 中, 下列代码段允许按钮注册一个 action 事件的是 ( )。(选择一项)
- a) button.enableActionEvents();
  - b) button.addActionListener(anActionListener);
  - c) button.enableEvents(true);
  - d) button.enableEvents(AWTEvent.ACTION\_EVENT\_MASK);

【解析】B

向组件注册事件监听器的方法是 addXXXListener。其中 XXX 代表事件。

事件监听器(event listener)就是一个实现 listener 接口的对象。所以, 程序员要做的就是创建一个 listener 对象, 然后向发起事件的组件注册这个对象。注册的过程就是调用组件的 addXXXListener()方法, 这里"XXX"表示组件所发起的事件的类型。只要看一眼"addListener"方法的名字就能知道组件能处理哪种事件了, 所以如果你让它听错了事件, 那么编译就根本通不过。到后面你就会看到, JavaBean 在决定它能处理哪些事件时, 也遵循"addListener"的命名规范。

- 18) 在 Java 语言中, 下面变量命名合法的有 ( )。(选择两项)
- a) variable123
  - b) 123variable
  - c) private
  - d) selg\_asd

【解析】AD

命名规范:

定义这个规范的目的是让项目中所有的文档都看起来像一个人写的, 增加可读性, 减少



项目组中因为换人而带来的损失。(这些规范并不是一定要绝对遵守,但是一定要让程序有良好的可读性)

- 普通变量 的命名

以字母或下划线开头,由字母、数字、下划线等组成。但不能是 java 的关键字。

- Package 的命名

Package 的名字应该都是由一个小写单词组成。

- Class 的命名

Class 的名字必须由大写字母开头而其他字母都小写的单词组成

- Class 变量的命名

变量的名字必须用一个小写字母开头。后面的单词用大写字母开头。

- Static Final 变量的命名

Static Final 变量的名字应该都大写,并且指出完整含义。

- 参数的命名

参数的名字必须和变量的命名规范一致。

- 数组的命名

数组应该总是用下面的方式来命名:

```
byte[] buffer;
```

而不是:

```
byte buffer[];
```

- 方法的参数

使用有意义的参数命名,如果可能的话,使用要和要赋值的字段一样的名字:

```
SetCounter(int size){  
    this.size = size;  
}
```

19) 在 JAVA 编程中,关于 TextField 的语句如下:

```
JTextField t = new JTextField("they are good",40);
```

下面 ( ) 是正确的。(选择两项)

- a) 在这个 JTextField 中,最多只能输入 40 个字符
- b) 在这个 JTextField 中,最少可以输入 40 个字符
- c) 在这个 JTextField 中,能够输入的字符不一定是 40 个
- d) 在这个 JTextField 中,用户可以编辑所输入的字符

【解析】CD

JTextField 构造方法的第二个参数表示文本框的列数。但并不限制它能输入的字符数。

```
public JTextField(String text,  
                  int columns)
```

创建一个新的文本域,并用待显示的指定文本初始化,且它的宽度足以显示指定的字符数。

参数:

text - 显示的文本。

columns - 字符数

20) 在 Java 语言中,按“东,西,南,北,中”指定组件的位置的布局管理器是 ( )。(选择一项)

- a) FlowLayout

- b) GridLayout
- c) BorderLayout
- d) CardLayout
- e) GridBagLayout

【解析】C

Java 中的布局方式:

BorderLayout 边界布局方式: 使用“东”、“西”、“南”、“北”、“居中”来指定组件的位置

FlowLayout 流式(顺序)布局: 按照从左到右由上到下的方式排列组件。

GridLayout 网格布局。把组件按照网格来放置。

CardLayout 卡片布局。对象是容器的布局管理器。它将容器中的每个组件当作一个卡片来处理。在某一时间, 只有一个卡片是可见的, 容器象一个卡片堆栈一样工作。... CardLayout 定义了一系列方法, 来允许一个应用顺序地翻动这些卡片, 或显示一个指定的卡片。是最灵活的布局方案。

21) 在 Java 中允许创建多线程应用程序的接口是 ( )。(选择一项)

- a) Threadable
- b) Runnable
- c) Clonable
- d) 以上均不是

【解析】A

22) 在 Java 语言中, 把组件放在 BorderLayout 的 ( ) 区域时, 它会自动垂直调整大小, 但不是水平调整。(选择一项)

- a) North 或 South
- b) East 或 West
- c) Center
- d) North, South 或 Center

【解析】A

在 BorderLayout 布局方式时, 当把组件放在 North 或 South 区域时, 会自动水平调整组件的大小。相反当在 East 或 West 方位时, 会自动垂直调整组件的大小。当在 Center 方位时, 垂直和水平方位都会自动调整。

23) 在 JAVA 编程中, 以下 ( ) 命令能够将 Java 源文件转换为类文件。(选择一项)

- a) appletviewer
- b) java
- c) javac
- d) 以上所有选项都不正确

【解析】C

Javac 是用来编译 java 的源文件的命令。产生类文件, 也就是字节文件。

Java 执行经过编译后产生的类文件。

Appletviewer 是用来查看 java Applet 程序的工具。

24) 在 JAVA 编程中, Java 具有下列 ( ) 特点。(选择三项)

- a) 面向对象

- b) 跨平台
- c) 安全
- d) 集中式体系结构
- e) 可编译成机器代码

【解析】ABC

JAVA 语言是完全面向对象的、能够跨平台、高安全性的分布式体系结构。可以被虚拟机编译为字节码。

25) 在 Java 中，欲定义某类所在的包外的所有类都能访问这个类，则应用的关键字是 ( )。

(选择一项)

- a) protected
- b) private
- c) public

【解析】C

Public 访问修饰符，该类所在包内和包外的类都可以访问。最开放。

Protected 访问修饰符，只有该类所在包内或它的派生类才能访问。

Private 访问修饰符，只有该类所在包内的类才能访问。

26) 在 Java 中，根据你的理解，下列方法 ( ) 可能是类 Orange 的构造函数。(选择三项)

- a) Orange() {...}
- b) Orange(...) {...}
- c) Public void Orange() {...}
- d) Public Orange() {...}
- e) Public OrangeConstructor() {...}

【解析】ABD

构造方法要注意的几个地方：

名字与类名一样。可以带参数重载。

不能带返回类型，包括 void。

27) 在 Java 中，下列选项表示字符 "a" 值的是 ( )。(选择一项)

- a) 'a'
- b) "a"
- c) new Character(a)
- d) \000a

【解析】A

字符应用单引号表示 'a';

new Character(a)表示创建一个字符对象 a

\000a 用 ASCII 表示应为 \097

28) 在 Java 中，可以使线程运行的方法是 ( )。(选择一项)

- a) init();
- b) start();
- c) resume();
- d) sleep();

【解析】B

29) 在 JAVA 语言中, 下面是 main() 方法的部分代码:

```
Frame f=new Frame( "My Frame" );
```

```
f.setSize(100,100);
```

为在屏幕显示 f, 应增加的代码是 ( )。(选择一项)

a) f.appear();

b) f.setForeground();

c) f.setVisible();

d) f.enable();

【解析】C

框架 Frame 是一个顶层容器。在创建后, 要在屏幕上显示只需将其设为可见。故选 C。调用 setVisible 方法。

A 项错误, 没有该方法

B 项设置该窗体的前景色。

D 项使该窗体为可用。

30) JAVA 中, 对于 drawImage(image, x, y, width, height, this) 方法解释正确的是 ( )。(选择两项)

a) image 是要绘制的图像

b) x, y 是表示图像的中心的位置

c) width 是源图像的宽度

d) this 是容器

【解析】A D

在指定位置并且按指定大小绘制指定的 Image。x,y 是表示要画图像的位置坐标。

31) JAVA 中, 欲返回按钮的标签, 其方法是 ( )。(选择一项)

a) getActionCommand()

b) setLabel(string str)

c) button()

d) getLabel()

【解析】A

欲返回按钮的标签使用 getActionCommand() 方法。

32) 在 JAVA 中, 假设我们有一个实现 ActionListener 接口的类, 以下方法, ( ) 能够为 一个 Button 类注册这个类。(选择一项)

a) addListener()

b) addActionListener()

c) addButtonListener()

d) setListener()

【解析】B

Swing 的事件模型中, 组件可以发起(或"射出")事件[译注 1]。各种事件都是类。当有事件发生时, 一个或多个"监听器(listener)"会得到通知, 并做出反应。这样事件的来源就同它的处理程序分隔开来了。一般说来, 程序员是不会去修改 Swing 组件的, 他们写的都是些事件处理程序, 当组件收到事件[译注 1]时, 会自动调用这些代码, 因此 Swing 的事件模型

可称得上是将接口与实现分隔开来的绝好范例了。

事件监听器(event listener)就是一个实现 listener 接口的对象。所以,程序员要做的就是创建一个 listener 对象,然后向发起事件的组件注册这个对象。注册的过程就是调用组件的 addXXXListener()方法,这里"XXX"表示组件所发起的事件的类型。只要看一眼"addListener"方法的名字就能知道组件能处理哪种事件了,所以如果你让它听错了事件,那么编译就根本通不过。到后面你就会看到,JavaBean 在决定它能处理哪些事件时,也遵循"addListener"的命名规范。

33) 在 JAVA 语言中,包 pack1 的类 class1 中有成员方法:protected void method\_1() {...},private void method\_2() {...},public void method\_3() {...}和 void method\_4() {...},在包 pack2 中的类 class2 不是 class1 的子类,你在 class2 中可以调用的方法有()。(选择一项)

- a) method\_1
- b) method\_2
- c) method\_3
- d) method\_4

【解析】C

本题可以通过画出包中类的结构来分析:

pack1: class1: 受保护的 method\_1 method\_4 私有的 method\_2 公共的 method\_3();

pack2: class2:

由于 class2 和 class1 之间没有继承关系;故 class2 只能调用 class1 种的 public 修饰的方法:method\_3()。

34) JAVA 中,使用()修饰符时,一个类能被同一包或不同包中的其他类访问。(选择一项)

- a) private
- b) protected
- c) public
- d) friendly

【解析】C

Public 访问修饰符,该类所在包内和包外的类都可以访问。最开放。

Protected 访问修饰符,只有该类所在包内或它的派生类才能访问。

Private 访问修饰符,只有该类所在包内的类才能访问。

Java 中无 friendly 修饰符。

35) 在 JAVA 中,类 Worker 是类 Person 的子类,Worker 的构造方法中有一句"super()",该语句()。(选择一项)

- a) 调用类 Worker 中定义的 super()方法
- b) 调用类 Person 中定义的 super()方法
- c) 调用类 Person 的构造函数
- d) 语法错误

【解析】C

有些时候经常需要在子类中调用父类的构造方法,此时需用 super 方法来调用。

36) 在 JAVA 语言中, 下列语言 ( ) 可以画出一矩形框架, 其距左边界为 0 像素, 距上边界为 10 像素, 宽为 30 像素, 高为 40 像素。(选择一项)

- a) Graphics g=new Graphics();  
g.drawRect(10,0,30,40);
- b) Graphics g=new Graphics();  
g.drawRect(0,10,30,40);
- c) Graphics g =new Graphics();  
g.drawRect(30,40,10,0)
- d) Graphics g =new Grphics();  
g.drawRect(30,40,0,10);

【解析】B

Graphics 类的方法 drawRect 的语法如下:

```
public void drawRect(int x, int y, int width, int height)
```

绘制指定矩形的轮廓。矩形的左边和右边分别为 x 和 x+width。顶部边沿和底部边沿分别为 y 和 y+height。该矩形是使用图形上下文的当前颜色绘制的。

参数:

x - 将被绘制的矩形的 x 坐标。

y - 将被绘制的矩形的 y 坐标。

width - 将被绘制的矩形的宽度。

height - 将被绘制的矩形的高度

故本题选 B

37) 在 JAVA 中, 类 Animal 中的方法 printA() 定义如下:

```
public void printA() {  
    int a=10;  
    int result=10%3;  
    System.out.println(result);  
}
```

在类 Dog 中方法 printA() 定义如下:

```
public void printA() {  
    int a=10;  
    Systme.out.println(a/3);  
}
```

Dog 类的定义如下:

```
Class Dog extends Animal{...}
```

若有语句:

```
Animal animal=new Dog();  
animal.printA();
```

则这段代码输出为 ( )。(选择一项)

- a) 0
- b) 3.3333
- c) 2
- d) 3

【解析】B

本题关键要理解如何使父类的对象调用子类的方法。因为通常都是子类调用父类的方

法。

```
Animal animal=new Dog();
```

此处虽然声明的是父类 Animal 类型的对象，但在初始化时调用的是子类的构造方法。正是因为此，使得 animal 对象指向 Dog 类的引用。该对象 animal 在调用 printA () 方法时不是调用父类的方法，而是调用在子类中重写的 printA () 方法。故本题执行  $10/3=3.3333$ 。

38) 在 JAVA 中，下列 ( ) 代码段允许按钮注册一个 action 事件。(选择一项)

- a) button.enableActioinEvents();
- b) button.addActionListener(anActionListener);
- c) button.enableEvents(true);
- d) button.enableEvents(AWTEvent.ACTION\_EVENT\_MASK);

【解析】B

Swing 的事件模型中，组件可以发起(或"射出")事件[译注 1]。各种事件都是类。当有事件发生时，一个或多个"监听器(listener)"会得到通知，并做出反应。这样事件的来源就同它的处理程序分隔开来了。一般说来，程序员是不会去修改 Swing 组件的，他们写的都是些事件处理程序，当组件收到事件[译注 1]时，会自动调用这些代码，因此 Swing 的事件模型可称得上是将接口与实现分隔开来的绝好范例了。

事件监听器(event listener)就是一个实现 listener 接口的对象。所以，程序员要做的就是创建一个 listener 对象，然后向发起事件的组件注册这个对象。注册的过程就是调用组件的 addXXXListener() 方法，这里"XXX"表示组件所发起的事件的类型。只要看一眼"addListener"方法的名字就能知道组件能处理哪种事件了，所以如果你让它听错了事件，那么编译就根本通不过。到后面你就会看到，JavaBean 在决定它能处理哪些事件时，也遵循"addListener"的命名规范。

39) 在 JAVA 编程中，实现 Runnable 接口时必须实现的方法是 ( )。(选择一项)

- a) wait()
- b) run()
- c) stop()
- d) start()

【解析】B

这是在 JBuild 中的工程中的 main 方法。程序从此处开始运行，需要运行进程。此处调用了 Runnable 接口的 run 方法，并对该方法进行了重写。

```
public static void main(String[] args) {  
    SwingUtilities.invokeLater(new Runnable() {  
        public void run() {  
            try  
            { UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());  
            } catch (Exception exception) {  
                exception.printStackTrace();  
            }  
        }  
    })  
}
```

Runnable 接口有一个最重要的方法 run，在进程要运行时，必须实现该方法。

40) 在 JAVA 语言中，下面关于类的描述正确的是 ( )。(选择一项)

- a) 一个子类可以有多个超类
- b) 一个超类可以有多个子类
- c) 子类可以使用超类的所有

【解析】B

一个超类可以派生出多个子类。

41) Java Applet 的三种状态: 1. Init(), 2. Start(), 3. Paint(), 在 Applet 载入时的顺序是 ( )。(选择一项)

- a) 1, 2, 3  
b) 2, 1, 3  
c) 2, 3, 1  
d) 1, 3, 2

【解析】A

- a) 过程
- b) 对象
- c) 组件
- d) 服务

【解析】B

- a) public
- b) class
- c) extends
- d) implements

【解析】C

A extends B

a) //注释  
b) /\*注释\*/  
c) /\*\*注释\*/  
d) /注释/

【解析】D

### 对单行代码注释

//注释

## 对多行代码注释

/\*注释\*/

在代码开头部分注释一些版权信息 `/**注释*/`



45) JAVA 线程编程中, 如果让线程睡眠, 可以用 ( ) 方法实现。(选择一项)

- a) start()
- b) close();
- c) setDaemon();
- d) sleep();

【解析】D

46) 在 JAVA 编程中, JPanel 缺省的布局管理器是 ( )。(选择一项)

- a) 该组件没有缺省的布局管理器
- b) FlowLayout
- c) JFlowLayout
- d) BorderLayout

【解析】D

```
47) import java.awt.*;
import java.applet.*;
public class DrawRect extends Applet
{
    public void paint(Graphics g)
    {
        g.setColor(Color.red);
        g.drawRoundRect(150, 50, 50, 50, 20, 40);
        g.setColor(Color.green);
        g.fillRoundRect(150, 140, 50, 50, 20, 20);
    }
}
```

关于上面 JAVA 代码的输出结果, 叙述正确的有 ( )。(选择两项)

- a) 两个圆角矩形的绘制弧高都为 20
- b) 两个圆角矩形的绘制弧宽相同
- c) 两个圆角矩形是左对齐的
- d) 两个圆角矩形是上对齐的

【解析】B C

Graphics 类的 drawRoundRect 方法用于绘制圆角的矩形。语法如下:

```
public abstract void drawRoundRect(int x, int y, int width, int height, int arcWidth, int arcHeight)
```

使用该图形上下文的当前颜色绘制轮廓为圆角的矩形。矩形的左边和右边分别为 x 和 x + width。矩形的顶部边沿和底部边沿分别为 y 和 y + height。

参数:

x - 将被绘制的矩形的 x 坐标。

y - 将被绘制的矩形的 y 坐标。

width - 将被绘制的矩形的宽度。

height - 将被绘制的矩形的高度。

arcWidth - 位于四个角上的弧的水平直径。 代表弧宽

arcHeight - 位于四个角上的弧的垂直直径 代表弧高

故本题选 B C

48) 在 JAVA 语言中, 不能被修改的变量是用关键字 ( ) 来修饰的。(选择一项)

- a) final
- b) class
- c) system
- d) void

【解析】A

49) 在 JAVA 编程中, ( ) 可以实现跳转结构。(选择一项)

- a) break
- b) while
- c) do-while
- d) for

【解析】A

Break 语句用于实现中止正在执行的循环, 从而跳出该循环, 转到其他语句执行。

50) 在 JAVA SWING 编程中, 创建一个窗体使用组件 ( )。(选择一项)

- a) JFrame
- b) INT
- c) CHAR
- d) LONG

【解析】A

Swing 是由 100% 纯 Java 实现的, Swing 组件是用 Java 实现的轻量级 (light-weight) 组件, 没有本地代码, 不依赖操作系统的支持, 这是它与 AWT 组件的最大区别。由于 AWT 组件通过与具体平台相关的对等类 (Peer) 实现, 因此 Swing 比 AWT 组件具有更强的实用性。Swing 在不同的平台上表现一致, 并且有能力提供本地窗口系统不支持的其它特性。在 AWT 包中创建窗体是 Frame 类, 为了区分, swing 包中的许多类都在前面加了一个 J 字母。故为 JFrame 类。

51) 在 JAVA 编程中, 编写一个 APPLET 需要继承 ( )。(选择一项)

- a) JFRAME
- b) APPLET
- c) THREAD
- d) FRAME

【解析】B

APPLET 是 Java 小程序, 专门用在网页中运行。通常需要继承 Applet 类。

52) 分析下列 java 代码:

```
class A
{
    public static void main(String[] args)
    {
        method();
    }
    static void method()
```

```

    {
        try
        {
            System.out.println( "Hello" );
        }
        finally
        {
            System.out.println( "good-bye" );
        }
    }
}

```

编译运行后，输出结果是（）。（选择一项）

- a) " Hello"
- b) " good-bye"
- c) "Hello"  
"good-bye"
- d) 代码不能编译

【解析】C

Java 中的异常处理模型如下：

```

    Try
    {
    }
} catch ( ) {}
Finally {}

```

其中 try 不能单独使用，必须和 catch 一起使用。finally 也不能单独使用。如果有 Finally 部分，则不管是否捕捉到了错误程序都将会执行它。

53) 下面选项中，（）可以用来在 HTML 中嵌入 APPLET 程序。（选择一项）

- a) <applet>...</applet>
- b) <title></titile>
- c) <br></br>
- d) <font></font>

【解析】A

在 HTML 中要嵌入 java 小程序，使用<applet>.....</applet>标记。

54) JAVA 中，按下和释放鼠标按钮的操作处理（）事件。（选择一项）

- a) mouseEntered
- b) mouseExited
- c) mousePressed
- d) mouseClicked

【解析】D

## Java 中的鼠标和键盘事件

Java 中的鼠标和键盘事件

1、使用 MouseListener 借口处理鼠标事件

鼠标事件有 5 种：按下鼠标键，释放鼠标键，点击鼠标键，鼠标进入和鼠标退出

鼠标事件类型是 MouseEvent，主要方法有：

getX(),getY() 获取鼠标位置

getModifiers() 获取鼠标左键或者右键

getClickCount() 获取鼠标被点击的次数

getSource() 获取鼠标发生的事件源

事件源获得监视器的方法是 addMouseListener(), 移去监视器的方法是 removeMouseListener()

处理事件源发生的时间的事件的接口是 MouseListener 接口中有如下的方法

mousePressed(MouseEvent) 负责处理鼠标按下事件

mouseReleased(MouseEvent) 负责处理鼠标释放事件

mouseEntered(MouseEvent) 负责处理鼠标进入容器事件

mouseExited(MouseEvent) 负责处理鼠标离开事件

mouseClicked(MouseEvent) 负责处理点击事件（包含了按下和释放鼠标按钮的操作）

2、使用 MouseMotionListener 接口处理鼠标事件

事件源发生的鼠标事件有 2 种：拖动鼠标和鼠标移动

鼠标事件的类型是 MouseEvent

事件源获得监视器的方法是 addMouseMotionListener()

处理事件源发生的事件的接口是 MouseMotionListener 接口中有如下的方法

mouseDragged() 负责处理鼠标拖动事件

mouseMoved() 负责处理鼠标移动事件

3、控制鼠标的指针形状

setCursor(Cursor.getPredefinedCursor(Cursor.鼠标形状定义)) 鼠标形状定义见（书 P 210）

4、键盘事件

键盘事件源使用 addKeyListener 方法获得监视器

键盘事件的接口是 KeyListener 接口中有 3 个方法

public void keyPressed(KeyEvent e) 按下键盘按键

public void keyReleased(KeyEvent e) 释放键盘按键

public void keyTyped(KeyEvent e) 按下又释放键盘按键

```
55) import java.applet.Applet;
    import java.awt.*;
    public class ImageDemo extends Applet
    {
        Image img;
        public void init()
        {
            img=getImage(getCodeBase(),"11.gif"); //1
        }
        public void paint(Graphics g)
        {
            int w=img.getWidth(this);
            int h=img.getHeight(this);
            g.drawImage(img,120,60,w/2,h/2,this); //2
            g.drawImage(img,150,0,w*2,h*2,this); //3
        }
    }
```

```
}  
}
```

以上 JAVA 代码第 ( ) 行将图像放大。(选择一项)

- a) 1
- b) 2
- c) 3
- d) 代码并没有将图像放大

【解析】C

方法: drawImage(image,x,y,width,height):在指定位置并且按指定大小绘制指定的 Image。x,y 是表示要画图像的位置坐标。

56) MyProgram.java 被编译后, 生成 ( )。(选择一项)

- a) MyProgram. Obj
- b) MyProgram. class
- c) MyProgram. exe
- d) MyProgram. bat

【解析】B

Java 源代码 (.java) 文件通过 javac 命令编译成.class 的字节文件。文件名不变。

57) JAVA 中, 访问修饰符限制性最高的是 ( )。(选择一项)

- a) private
- b) protected
- c) public
- d) friendly

【解析】A

Public 访问修饰符, 该类所在包内和包外的类都可以访问。最开放。

Protected 访问修饰符, 只有该类所在包内或它的派生类才能访问。

Private 访问修饰符, 只有该类所在包内的类才能访问。限制性最高。

Java 中没有 friendly 修饰符。

58) 在 JAVA 编程中, 关于 Swing 包中的组件, 下面 ( ) 是正确的。(选择一项)

- a) Swing 中的每个组件都是采用 MVC 模式设计的
- b) JFrame 窗口的关闭按钮默认不能使窗口关闭
- c) Swing 的组件和 awt 组件, 在编程时不能混合使用
- d) 以上都正确

【解析】A

Swing 组件的设计都采用了 MVC 模式 (Model/View/Control);

JFrame 窗口的关闭按钮默认能使窗口关闭

通常可以将 swing 的组件和 awt 的组件可以混合使用。

59) 在 Java 语言中, 下列组件可以让用户选择多个选项有 ( )。(选择两项)

- a) Checkbox
- b) Radiobutton
- c) List
- d) Choice

【解析】A C

Checkbox 为复选框按钮, 可以让用户选择多个选项。

Radiobutton 单选按钮，每次只能选择一个选项。

List 为列表组件，可以让用户选择多个选项。

没有 D 项这样的组件。

60) 在 JAVA SWING 编程中，创建一个窗体使用组件 ( )。(选择一项)

- a) JFRAME
- b) INT
- c) CHAR
- d) LONG

【解析】

61) 分析下列 java 代码：

```
class A
{
    public static void main(String[] args)
    {
        method();
    }
    static void method()
    {
        try
        {
            System.out.println( "Hello" );
        }
        finally
        {
            System.out.println( "good-bye" );
        }
    }
}
```

编译运行后，输出结果是 ( )。(选择一项)

- a) " Hello"
- b) " good-bye"
- c) "good-bye"
- d) 代码不能编译

【解析】

62) 下面选项中，( ) 可以用来在 HTML 中嵌入 APPLET 程序。(选择一项)

- a) <applet>..</applet>
- b) <title></titile>
- c) <br></br>
- d) <font></font>

【解析】

63) JAVA 中, 按下和释放鼠标按钮的操作处理 ( ) 事件。(选择一项)

- a) mouseEntered
- b) mouseExited
- c) mousePressed
- d) mouseClicked

【解析】

64) 在 JAVA 编程中, 分析下面的 Java 代码:

```
import java.net.*;
class hello implements Runnable
{
    Public static void main (String args[])
    {
        Thread t=new Thread (this);
        t.start ();
    }
    Public void run ()
    {
        System.out.println ("你好");
    }
}
```

以下说法正确的是 ( )。(选择一项)

- a) 此程序不能通过编译
- b) 该程序运行时没有任何输出结果
- c) 输出“你好”
- d) 该程序能够通过编译, 但运行时出错

【解析】

65) 下面描述 ( ) 是正确的。(选择两项)

- a) Java 的源程序必须以“.java”或“.jav”扩展名保存
- b) Java VM 可以是软件也可以是硬件
- c) Java 使用解释器执行代码
- d) 用高级语言编写的代码可以让计算机理解并执行它们

【解析】

66) 下面哪些类属于 java.lang 包中的类 ( ) (选择二项)

- a) StringBuffer
- b) Date
- c) Math
- d) Exception

【解析】





## 《面向对象程序设计 JAVA 语言程序设计》期末考试试题及部分答案2

### 一、单项选择题(每小题2分,共10分)

1、编译 Java Application 源程序文件将产生相应的字节码文件,这些字节码文件的扩展名为( B )。

- A. .java
- B. .class
- C. .html
- D. .exe

2、设  $x=1, y=2, z=3$ , 则表达式  $y+=z--/++x$  的值是(A )。

- A. 3
- B. 3.5
- C. 4
- D. 5

3、在 Java Applet 程序用户自定义的 Applet 子类中,一般需要重载父类的( D )方法来完成一些画图操作。

- A. start()
- B. stop()
- C. init()
- D. paint()

4、不允许作为类及类成员的访问控制符的是( C )。

- A. public
- B. private
- C. static
- D. protected

5、为 AB 类的一个无形参无返回值的方法 method 书写方法头,使得使用类名 AB 作为前缀就可以调用它,该方法头的形式为( A )。

- A. static void method()
- B. public void method()
- C. final void method()
- D. abstract void method()

### 二、填空题(每空格1分,共20分)

1、开发与运行 Java 程序需要经过的三个主要步骤为 编辑源程序、编译生成字节码、和 解释运行字节码。

2、如果一个 Java Applet 源程序文件只定义有一个类,该类的类名为 MyApplet,则类 MyApplet 必须是 Applet、类的子类并且存储该源程序文件的文件名为 MyApplet。

3、如果一个 Java Applet 程序文件中定义有3个类,则使用 Sun 公司的 JDK 编译器 javac.exe 编译该源程序文件将产生 3 个文件名与类名相同而扩展名为 .class 的字节码文件。

4、在 Java 的基本数据类型中, char 型采用 Unicode 编码方案,每个 Unicode 码占用 2 字节内存空间,这样,无论是中文字符还是英文字符,都是占用 2 字节内存空间。

5、设  $x=2$ , 则表达式  $(x++)/3$  的值是 0。

6、若  $x=5, y=10$ , 则  $x < y$  和  $x \geq y$  的逻辑值分别为 true 和 false。

7、 抽象(abstract)方法 方法是一种仅有方法头, 没有具体方法体和操作实现的方法, 该方法必须在抽象类之中定义。 最终(final)方法 方法是不能被当前类的子类重新定义的方法。

8、创建一个名为 MyPackage 的包的语句是 package MyPackage;

该语句应该放在程序的位置为: 应该在程序第一句。

9、设有数组定义: int MyIntArray[] = { 10, 20, 30, 40, 50, 60, 70}; 则执行以下几个语句后的输出结果是 120。

```
int s = 0;
for ( int i = 0 ; i < MyIntArray.length ; i ++ )
    if ( i % 2 == 1 )    s += MyIntArray[i];
System.out.println( s );
```

10、在 Java 程序中, 通过类的定义只能实现 单 重继承, 但通过接口的定义可以实现 多 重继承关系。

三、写出下列程序完成的功能。(每小题5分, 共20分)

1、 public class Sum

```
{ public static void main( String args[ ])
{ double sum = 0.0;
  for ( int i = 1 ; i <= 100 ; i ++ )
    sum += 1.0/(double)i;
  System.out.println( "sum="+sum );
}
```

2、 import java.io.\*;

```
public class Reverse
{ public static void main(String args[ ])
{ int i, n=10;
  int a[ ] = new int[10];
  for ( i = 0 ; i < n ; i ++ )
  try {
    BufferedReader br = new BufferedReader(
      new InputStreamReader(System.in));
    a[i] = Integer.parseInt(br.readLine()); // 输入一个整数
  } catch ( IOException e ) { };
  for ( i = n-1 ; i >= 0 ; i -- )
    System.out.print(a[i]+" ");
}
```

```

        System.out.println( );
    }
}

3、 import    java.awt.*;

    public    class    abc

    { public    static    void    main(String args[])

        {        new FrameOut();    }

    }

    class    FrameOut    extends    Frame    // Frame 为系统定

    {    Button btn;                                // 义的窗框类

        FrameOut( )

        {        super("按钮");

                btn = new Button("按下我");

                setLayout(new    FlowLayout( ));

                add(btn);

                setSize(300,200);

                show( );

        }

    }

4、 import    java.io.*;

    public    class    abc

    {    public    static    void    main(String args[])

        {    SubClass    sb = new    SubClass( );

            System.out.println(sb.max( ));

        }

    }

    class    SuperClass

    {    int a = 10 , b = 20 ; }

    class SubClass extends SuperClass

    {    int max( ) { .return    ((a>b)?a:b); } }

```

四、写出下面程序的运行结果(每小题10分，共30分)

```

1、 import    java.io.*;

    public class abc

    {        public static void main(String args{ })

```

```

        {   AB s = new AB("Hello!", "I love JAVA.");
            System.out.println(s.toString());
        }
    }

```

```

class AB {
    String s1;
    String s2;

    AB( String str1 , String str2 )
    { s1 = str1; s2 = str2; }

    public String toString()
    { return s1+s2;}
}

```

```

2. import java.io.*;

public class abc
{
    public static void main(String args[] )
    {
        int i, s = 0;

        int a[] = { 10, 20, 30, 40, 50, 60, 70, 80, 90 };

        for ( i = 0; i < a.length; i++)
            if ( a[i]%3 == 0 ) s += a[i];

        System.out.println("s="+s);
    }
}

```

```

3. import java.io.*;

public class abc
{
    public static void main(String args[] )
    {
        { System.out.println("a="+a+"\nb="+b); }
    }

    class SubClass extends SuperClass
    { int c;

        SubClass(int aa,int bb,int cc)
        { super(aa,bb);

```

```

        c=cc;
    }
}

class SubSubClass extends SubClass
{
    int a;

    SubSubClass(int aa,int bb,int cc)
    {
        super(aa,bb,cc);
        a=aa+bb+cc;
    }

    void show()
    { System.out.println("a="+a+"\nb="+b+"\nc="+c); }
}

```

##### 五、使用 Java 语言编写程序。(每小题10分，共20分)

1、编写一个字符界面的 Java Application 程序，接受用户输入的10个整数，并输出这10个整数的最大值和最小值。

2、编写一个完整的 Java Applet 程序使用复数类 Complex 验证两个复数  $1+2i$  和  $3+4i$  相加产生一个新的复数  $4+6i$ 。

复数类 Complex 必须满足如下要求：

(1) 复数类 Complex 的属性有：

RealPart: int 型，代表复数的实数部分

ImaginPart: int 型，代表复数的虚数部分

(2) 复数类 Complex 的方法有：

Complex(): 构造函数，将复数的实部和虚部都置0

Complex( int r, int i ): 构造函数，形参 r 为实部的初值，i 为虚部的初值。

Complex complexAdd(Complex a): 将当前复数对象与形参复数对象相加，所得的结果仍是一个复数值，返回给此方法的调用者。

String ToString(): 把当前复数对象的实部、虚部组合成  $a+bi$  的字符串形式，其中 a 和 b 分别为实部和虚部的数据。

##### 《JAVA 语言程序设计》期末考试模拟试题

##### 参考答案及评分标准

##### 一、单选题(每小题2分，共10分)

1、B      2、A      3、D      4、C      5、A

二、填空题（每空格1分，共20分）

1、编辑源程序、编译生成字节码、解释运行字节码

2、Applet、MyApplet

3、javac.exe、3、.class

4、2、2

5、0

6、true、false

7、抽象(abstract)方法、最终(final)方法

8、package    MyPackage;      应该在程序第一句。

9、120

10、单、多

三、写出下列程序完成的功能。（每小题5分，共20分）

1、计算  $1/1+1/2+1/3+\dots+1/100$  的值。

2、从标准输入(即键盘)读入10个整数存入整型数组 a 中，然后逆序输出这10个整数。

3、创建一个标题为"按钮"的窗框，窗框中显示有"按下我"字样的按钮。

4、求两个数的最大值。

四、写出下面程序的运行结果(每小题10分，共30分)

1、Hello! I love JAVA.

2、s = 180

3、a=60

b=20

c=30

五、使用 Java 语言编写程序。（每小题10分，共20分）

1、参考程序如下：

```
import java.io.*;

public class abc
{
    public static void main(String args[] )
    { int i, n = 10, max = 0, min = 0, temp = 0;
      try {
          BufferedReader br = new BufferedReader(
              new InputStreamReader(System.in));
```

```

        max = min = Integer.parseInt(br.readLine( ));
    } catch ( IOException e ) { } ;
    for ( i = 2 ; i <= n ; i ++ ) {
        try {
            BufferedReader br = new BufferedReader(
                new InputStreamReader(System.in));
            temp = Integer.parseInt(br.readLine( ));
            if (temp > max ) max=temp;
            if (temp < min) min=temp;
        } catch ( IOException e ) { } ;
    }
    System.out.println("max="+max+"\nmin="+min);
}
}

```

2、参考程序如下:

```

import java.applet.* ;
import java.awt.* ;
public class abc extends Applet
{
    Complex a,b,c;
    public void init( )
    {
        a = new Complex(1,2);
        b = new Complex(3,4);
        c = new Complex();
    }

    public void paint(Graphics g)
    {
        c=a.complexAdd(b);
        g.drawString("第一个复数: "+a.toString(),10,50);
        g.drawString("第二个复数: "+b.toString(),10,70);
        g.drawString("两复数之和: "+c.toString(),10,90);
    }
}

```

```
class Complex
```

```
{  
    int RealPart;    // 复数的实部  
    int ImaginPart; // 复数的虚部  
    Complex() { RealPart = 0; ImaginPart = 0; }  
    Complex(int r, int i)  
    { RealPart = r;    ImaginPart = i;    }  
    Complex complexAdd(Complex a)  
    {  
        Complex temp = new Complex(); // 临时复数对象  
        temp.RealPart = RealPart + a.RealPart;  
        temp.ImaginPart = ImaginPart + a.ImaginPart;  
        return temp;  
    }  
    public String toString()  
    {    return ( RealPart + " + " + ImaginPart + " i ");    }  
}
```