


```

5.         public static void main (String args[])
6.         {
7.             System.out.println( "x=" + x);
8.         }
9.         static { x/=3;}
10.    }

```

- A、4 行与 9 行不能通过编译，因为缺少方法名和返回类型
- B、9 行不能通过编译，因为只能有一个静态初始化器
- C、编译通过，执行结果为：x=5
- D、编译通过，执行结果为：x=3

12. 关于以下程序代码的说明正确的是 ()

```

1. class HasStatic{
2.     private static int x=100;
3.     public static void main(String args[]){
4.         HasStatic hs1=new HasStatic();
5.         hs1.x++;//101
6.     HasStatic hs2=new HasStatic();
7.         hs2.x++;//102
8.         hs1=new HasStatic( );
9.         hs1.x++;//103
10.        HasStatic.x-;//102
11.        System.out.println( "x=" +x);
12.    }
13. }

```

- A、5 行不能通过编译，因为引用了私有静态变量
- B、10 行不能通过编译，因为 x 是私有静态变量
- C、程序通过编译，输出结果为：x=103
- D、程序通过编译，输出结果为：x=102

13. 以下选项中循环结构合法的是 ()

- A、while (int i<7)


```

      {    i++;
        System.out.println("i is "+i);
      }

```
- B、int j=3;


```

      while(j)
      { System.out.println(" j is "+j);
      }

```
- C、int j=0;


```

      for(int k=0; j + k !=10; j++,k++)
      {    System.out.println(" j is "+ j + "k is"+ k);
      }

```
- D、int j=0;


```

      do{
        System.out.println( "j is "+j++);

```

```
        if (j == 3) {continue loop;}  
    }while (j<10);
```

14. 类 Test1 定义如下:

```
1. public class Test1{  
2.     public float aMethod (float a, float b) {}  
3.  
4. }
```

将以下哪种方法插入行 3 是不合法的。()

- A、public float aMethod (float a, float b, float c) { }
- B、public float aMethod (float c, float d) { }
- C、public int aMethod (int a, int b) { }
- D、private float aMethod (int a, int b, int c) { }

15. 类 Test1、Test2 定义如下:

```
1. public class Test1  
2. {public float aMethod (float a, float b) throws  
3.     IOException{  
4.     }  
5. public class Test2 extends Test1{  
6.  
7. }
```

将以下哪种方法插入行 6 是不合法的。()

- A、float aMethod (float a, float b) {}
- B、public int aMethod (int a, int b) throws Exception{ }
- C、public float aMethod (float p, float q) {}
- D、public int aMethod (int a, int b) throws IOException{ }

16. 关于以下程序段, 正确的说法是 ()

```
1. String s1=" abc" + " def" ;  
2. String s2=new String (s1);  
3. if (s1==s2)  
4.     System.out.println( " == succeeded" );  
5. if (s1.equals(s2))  
6.     System.out.println( ".equals() succeeded" );
```

- A、行 4 与行 6 都将执行
- B、行 4 执行, 行 6 不执行
- C、行 6 执行, 行 4 不执行
- C、行 4、行 6 都不执行

17. Java application 中的主类需包含 main 方法, main 方法的返回类型是什么? ()

- A、int
- B、float
- C、double
- D、void

18. 以下哪个方法用于定义线程的执行体? ()

- A、start()
- B、init()
- C、run()
- D、synchronized()

19. 以下标识符中哪项是不合法的()

- A、const
- B、\$double
- C、hello
- D、BigMeaninglessName

20. 以下哪个关键字可以用来为对象加互斥锁? ()

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

21. 在 Java 中，一个类可同时定义许多同名的方法，这些方法的形式参数个数、类型或顺序各不相同，传回的值也可以不相同。这种面向对象程序的特性称为（ ）。

A、隐藏 B、覆盖 C、重载 D、Java 不支持此特性

22. 以下关于构造函数的描述错误的是（ ）。

- A、构造函数的返回类型只能是 void 型。
- B、构造函数是类的一种特殊函数，它的方法名必须与类名相同。
- C、构造函数的主要作用是完成对类的对象的初始化工作。
- D、一般在创建新对象时，系统会自动调用构造函数。

23. 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

24. 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

25. 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

26. 有以下程序片段，下列哪个选项不能插入到行 1。（ ）

```
public class Interesting{  
    //do sth  
}
```

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

27. 以下哪项是接口的正确定义？（ ）

- A、interface B{ void print() { };}
- B、abstract interface B{ void print() ;}
- C、abstract interface B extends A1,A2 //A1、A2 为已定义的接口
{ abstract void print(){ };}
- D、interface B{ void print();}

28. 下面哪个函数是 public void aMethod(){...}的重载函数？（ ）

- A、void aMethod(){...}
- B、public int aMethod(){...}
- C、public void aMethod (){...}
- D、public int aMethod (int m){...}

29. A 派生出子类 B，B 派生出子类 C，并且在 Java 源代码中有如下声明：

1. A a0=new A();
2. A a1=new B();
3. A a2=new C();

问以下哪个说法是正确的？（ ）


```

1. class A{
2.     protected boolean equals(){
3.         return super.equals();}
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
37. 有语句 `String s=" hello world"` ; , 以下操作哪个是不合法的? ()
A、`int i=s.length();` B、`s>>>=3;` C、`String ts=s.trim();` D、`String t=s+" !"` ;
38. 下列关于 Java 语言的特点，描述错误的是 ()
A. Java 是跨平台的编程语言 B. Java 支持分布式计算
C. Java 是面向过程的编程语言 D. Java 支持多线程
39. 下述概念中不属于面向对象方法的是 ()。
A. 对象、消息 B. 继承、多态 C. 类、封装 D. 过程调用
40. 结构化程序设计所规定的三种基本控制结构是 ()
A. 输入、处理、输出 B. 树形、网形、环形
C. 顺序、选择、循环 D. 主程序、子程序、函数
41. 下列关于构造方法的叙述中，错误的是 ()
A. Java 语言规定构造方法名与类名必须相同
B. Java 语言规定构造方法没有返回值，但不用 `void` 声明
C. Java 语言规定构造方法不可以重载
D. Java 语言规定构造方法只能通过 `new` 自动调用
42. 下列哪个类的声明是正确的? ()
A. `abstract final class HI{` B. `abstract private move(){}`
C. `protected private number;` D. `public abstract class Car{`
43. 关于被私有访问控制符 `private` 修饰的成员变量，以下说法正确的是 ()
A. 可以被三种类所引用：该类自身、与它在同一个包中的其他类、在其他包中的该类的子类
B. 可以被两种类访问和引用：该类本身、该类的所有子类
C. 只能被该类自身所访问和修改
D. 只能被同一个包中的类访问
44. 下列关于 `for` 循环和 `while` 循环的说法中哪个是正确的? ()
A. `while` 循环能实现的操作，`for` 循环也都能实现
B. `while` 循环判断条件一般是程序结果，`for` 循环判断条件一般是非程序结果
C. 两种循环任何时候都可替换
D. 两种循环结构中都必须有循环体，循环体不能为空
45. 阅读以下代码：
- ```

import java.io.*;
import java.util.*;
public class foo{
 public static void main (String[] args){
 String s;
 System.out.println("s=" + s);

```

```
 }
}
```

输出结果应该是：（ ）

- A. 代码得到编译，并输出 “s=”
- B. 代码得到编译，并输出 “s=null”
- C. 由于 String s 没有初始化，代码不能编译通过
- D. 代码得到编译，但捕获到 NullPointerException 异常

46. 编译运行以下程序后，关于输出结果的说明正确的是（ ）

```
public class Conditional{
 public static void main(String args[]){
 int x=4;
 System.out.println("value is "+ ((x>4) ? 99.9:9));
 }
}
```

- A. 输出结果为：value is 99.9
- B. 输出结果为：value is 9
- C. 输出结果为：value is 9.0
- D. 编译错误

47. 执行完以下代码 `int [] x = new int[10];` 后，以下哪项说明是正确的（ ）

- A. x[9]为0
- B. x[9]未定义
- C. x[10]为0
- D. x[0]为空

48. 关于以下程序段，正确的说法是（ ）

```
1. String s1= "a" + "b"
2. String s2=new String (s1);
3. if (s1==s2)
4. System.out.println("= = is succeeded");
5. if (s1.equals(s2))
6. System.out.println(".equals() is succeeded");
```

- A. 行 4 与行 6 都将执行
- B. 行 4 执行，行 6 不执行
- C. 行 6 执行，行 4 不执行
- C. 行 4、行 6 都不执行

49. 以下程序的运行结果是：（ ）

```
public class Increment{
 public static void main(String args[]){
 int c;
 c = 2;
 System.out.println(c);
 System.out.println(c++);
 System.out.println(c);
 }
}
```

- A. 2 2 2
- B. 2 3 3
- C. 2 2 3
- D. 3 4 4

50. 下列哪一个关键字用于实现接口来定义类？（ ）

- A、 extends
- B、 implements
- C、 abstract
- D、 interface

51. 若 a 的值为 3 时，下列程序段被执行后，c 的值是多少？（ ）
- ```
if ( a>0 )
    if ( a>3 ) c = 2;
    else c = 3;
else c = 4;
```
- A、1 B、2 C、3 D、4
52. 下面哪一个操作符的优先级最高？（ ）
- A、&& B、|| C、! D、()
53. 下列方法头中哪一个不与其他方法形成重载(overload)关系？（ ）
- A、void mmm() B、void mmm(int i) C、void mmm(String s) D、int mm()
54. 编译 Java 程序的命令是：（ ）
- A、appletviewer B、javac C、java D、javadoc
55. 编译 Java 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、.byte B、.class C、.html D、.exe
56. 执行语句 int i = 1, j = ++i; 后 i 与 j 的值分别为（ ）。
- A、1 与 1 B、2 与 1 C、1 与 2 D、2 与 2
57. main 方法是 Java 应用程序执行的入口点，关于 main 方法的方法头以下哪项是合法的？（ ）
- A、public static void main () B、public static void main (String[] args)
- C、public static int main (String[] arg) D、public void main (String arg[])
58. 下列哪个选项不是 Java 语言的特点？（ ）
- A、面向对象 B、高安全性 C、平台无关 D、面向过程
59. 下列哪个是合法的 Java 标识符？（ ）
- A、Tree&Glasses B、FirstJavaApplet C、*theLastOne D、273.5
60. 下面的选项中，哪一项不属于“汽车类”的行为（ ）。
- A、启动 B、刹车 C、减速 D、速度
61. 下面哪一个循环会导致死循环？（ ）
- A、for (int k = 0; k < 0; k++) B、for (int k = 10; k > 0; k--)
- C、for (int k = 0; k < 10; k--) D、for (int k = 0; k > 0; k++)
62. 有如下程序段：
- ```
int a = b = 5;
String s1 = "祝你今天考出好成绩! ";
String s2 = s1;
```
- 则表达式 a == b 与 s2 == s1 的结果分别是：（ ）。
- A、true 与 true                      B、false 与 true                      C、true 与 false                      D、false 与 false
63. 在 Java 中用什么关键字修饰的方法可以直接通过类名来调用？（ ）
- A、static                      B、final                      C、private                      D、void
64. 若在某一个类定义中定义有如下的方法：abstract void performDial();该方法属于（ ）。
- A、接口方法                      B、最终方法                      C、抽象方法                      D、空方法
65. 有如下程序段：





```

 System.out.print(ex.str+" and ");
 Sytem.out.print(ex.ch);
 }
 public void change(String str,char ch[]){
 str="test ok";
 ch[0]='g';
 }
}

```

- A、 good and abc
- B、 good and gbc
- C、 test ok and abc
- D、 test ok and gbc

75. 运行下列程序, 会产生什么结果 ( )

```

public class X extends Thread implements Runnable{
 public void run(){
 System.out.println("this is run()");
 }
 public static void main(String args[]) {
 Thread t=new Thread(new X());
 t.start();
 }
}

```

- A、 第一行会产生编译错误
- B、 第六行会产生编译错误
- C、 第六行会产生运行错误
- D、 程序会运行和启动

76. 要从文件"file.dat"中读出第 10 个字节到变量 c 中,下列哪个方法适合? ( )

- A、 FileInputStream in=new FileInputStream("file.dat"); in.skip(9); int c=in.read();
- B、 FileInputStream in=new FileInputStream("file.dat"); in.skip(10); int c=in.read();
- C、 FileInputStream in=new FileInputStream("file.dat"); int c=in.read();
- D、 RandomAccessFile in=new RandomAccessFile("file.dat"); in.skip(9); int c=in.readByte();

77. Java 编程所必须的默认引用包为( B )

- A. java.sys 包
- B. java.lang 包
- C. java.util 包
- D. 以上都不是

78. 下面语句在编译时不会出现警告或错误的是( )

- A. float f=3.14;
- B. char c="c";
- C. Boolean b=null;
- D. int i=10.0;

79. 下面不是合法标识符的是( )

- A. 2ofUS
- B. giveMes
- C. whataQuiz
- D. \$d2000\_

80. 下面哪一个是合法的数组声明和构造语句( )

- A. int[] ages = [100];
- B. int ages = new int[100];
- C. int[] ages = new int[100];
- D. int() ages = new int(100);

81. 下面说法不正确的是( C )

- A. 一个子类的对象可以接收父类对象能接收的消息;
- B. 当子类对象和父类对象能接收同样的消息时, 它们针对消息产生的行为可能不同;
- C. 父类比它的子类的方法更多;
- D. 子类在构造函数中可以使用 `super( )`来调用父类的构造函数;

82. 给出下面代码段, 哪行将引起一个编译时错误? ( )

```

1) public class Test {
2) int n = 0;
3) int m = 0;
4) public Test(int a) { m=a; }
5) public static void main(String arg[]) {
6) Test t1,t2;
7) int j,k;
8) j=3; k=5;
9) t1=new Test();
10) t2=new Test(k);
11) }
12) }
```

- A. 行 1
- B. 行 4
- C. 行 6
- D. 行 9

83. 下面程序中类 `ClassDemo` 中定义了一个静态变量 `sum`, 分析程序段的输出结果。( )

```

class ClassDemo {
 public static int sum=1;
 public ClassDemo() {
 sum = sum + 5;
 }
}

public class ClassDemoTest{
 public static void main(String args[]) {
 ClassDemo demo1=new ClassDemo();
 ClassDemo demo2=new ClassDemo();
 System.out.println(demo1.sum);
 }
}
```

- A. 0
- B. 6
- C. 11
- D. 2

84. 下面这些类型的应用, 那个不使用 `Java` 语言来编写? ( )

- A) `JavaScript`
- B) `Applet`
- C) `Servlet`
- D) `Java Swing`

85. 声明成员变量时, 如果不使用任何访问控制符(`public`, `protected`, `private`), 则以下哪种类型的类不能对该成员进行直接访问 ( )

- A) 同一类
- B) 同一包中的子类
- C) 同一包中的非子类
- D) 不同包中的子类

86. 下列哪种异常是检查型异常, 需要在编写程序时声明 ( )

- A) `NullPointerException`
- B) `ClassCastException`
- C) `FileNotFoundException`
- D) `IndexOutOfBoundsException`

87. 下面哪个流类属于面向字符的输入流( )



- B) `new BufferedReader(new FileInputStream("a.dat"));`
- C) `new GZIPOutputStream(new FileOutputStream("a.zip"));`
- D) `new ObjectInputStream(new FileInputStream("a.dat"));`

93. Java 的集合框架中重要的接口 `java.util.Collection` 定义了许多方法。选项中哪个方法不是 `Collection` 接口所定义的? ( )

- A) `int size()`
- B) `boolean containsAll(Collection c)`
- C) `compareTo(Object obj)`
- D) `boolean remove(Object obj)`

94. 一个线程在任何时刻都处于某种线程状态 (thread state), 例如运行状态、阻塞状态、就绪状态等。一个线程可以由选项中的哪种线程状态直接到达运行状态? ( )

- A) 死亡状态
- B) 阻塞状态 (对象 lock 池内)
- C) 阻塞状态 (对象 wait 池内)
- D) 就绪状态

95. 选项中哪一行代码可以替换题目中 `//add code here` 而不产生编译错误? ( )

```
public abstract class MyClass {
 public int constInt = 5;
 //add code here
 public void method() {}
}
```

- A) `public abstract void method(int a);`
- B) `value = value + 5;`
- C) `public int method();`
- D) `public abstract void anotherMethod() {}`

96. `File` 类是 `IO` 包中唯一表示磁盘文件信息的对象, 它定义了一些与平台无关的方法来操纵文件。通过调用 `File` 类提供的各种方法, 我们能够创建、删除文件、重命名文件、判断文件的读写权限及是否存在, 设置和查询文件的最近修改时间等。下面的代码片段实现的是什么功能? ( )

```
File file = new File("C:\\test.dat");
if (file.exists()) {
 file.delete();
}
```

- A) 创建 `C:\test.dat`。
- B) 删除 `C:\test.dat`。
- C) 打开 `C:\test.dat` 文件输出流。
- D) 移动 `C:\test.dat`

97. 阅读 `Shape` 和 `Circle` 两个类的定义。在序列化一个 `Circle` 的对象 `circle` 到文件时, 下面哪个字段会被保存到文件中? ( )

```
class Shape {
 public String name;
}
class Circle extends Shape implements Serializable {
 private float radius;
 transient int color;
 public static String type = "Circle";
}
```

- A) name                      B) radius                      C) color                      D) type

98. 下面是 People 和 Child 类的定义和构造方法，每个构造方法都输出编号。在执行 new Child("mike")的时候都有哪些构造方法被顺序调用？请选择输出结果 ( )

```
class People {
 String name;
 public People() { System.out.print(1); }
 public People(String name) {
 System.out.print(2);
 this.name = name;
 }
}
class Child extends People {
 People father;
 public Child(String name) {
 System.out.print(3);
 this.name = name;
 father = new People(name + ":F");
 }
 public Child(){ System.out.print(4); }
}
```

- A) 312                      B) 32                      C) 432                      D) 132

99. 下面哪个选项中的代码没有定义内部类,或者错误的定义了内部类? ( )

- A) public Class Line { int length; Class Point { //内部类代码} }  
 B) public Class Line { public Point getPoint() { return new Point() { //内部类代码}; } }  
 C) public Class Line { //外部类代码 Class Point { //内部类代码 } }  
 D) public Class Line { public int calcLength() { Class Point { //内部类代码 } } }

100. list 是一个 ArrayList 的对象，哪个选项的代码填写到 //todo delete 处，可以在 Iterator 遍历的过程中正确并安全的删除一个 list 中保存的对象? ( )

```
Iterator it = list.iterator();
int index = 0;
while (it.hasNext()){
 Object obj = it.next();
 if (needDelete(obj)) { //needDelete 返回 boolean，决定是否要删除
 //todo delete
 }
 index ++;
}
```

- A) list.remove(obj);                      B) list.remove(index);  
 C) list.remove(it.next());                      D) it.remove();

|       |       |       |       |       |       |       |       |       |        |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| 1-5   | 6-10  | 11-15 | 16-20 | 21-25 | 26-30 | 31-35 | 36-40 | 41-45 | 46-50  |
| DADCC | BCBCB | CDCBA | CDCAD | CABAC | DDDDC | DBACD | DBCDC | CDCBC | CACCB  |
| 51-55 | 56-60 | 61-65 | 66-70 | 71-75 | 76-80 | 81-85 | 86-90 | 91-95 | 96-100 |
| CDDBB | DBDBD | CAACA | ABAAD | CBCBD | ABCAC | CDCAD | CACDD | ABCDA | BBBCD  |

## 二、填空题

1. 本题的功能是计算二维数组各个元素的和。程序中定义了二维数组 `arr`，`arr` 有 3 行 4 列共 12 个元素，程序中采用 `for` 循环语句的嵌套来计算数组中各个元素的和，并将结果保存在 `sum` 变量中，最后打印输出结果。

```
public class java1{
 public static void main(String[] args){
 int arr[][]={{1,2,3,4},{5,6,7,8},{9,10,11,12}};
 int sum=0;
 int i=0,j=0;
 for(i=0; _____)
 for(_____)
 _____;
 System.t.println("sum="+sum);
 }
}
```

参考答案：

第 1 处： `i<3;i++` 或 `i<=2;i++`

第 2 处： `j=0;j<4;j++` 或 `j=0; j=3; j++`

第 3 处： `sum=sum+arr[i][j]`

2. 若 `byte b=123; short s=456;` 则表达式 `b+s` 的类型是 int。

3. 数组 `x` 定义：`String x[][]=new int[3][2]; x[0][0]="abc",x[0][1]="12345";`  
则 `X.length` 的值为 6，`x[0][1].length()` 的值为 5。

4. Java 语言的变量由变量名、变量类型、变量属性、变量初值组成。

5. 在 Java 中，常数 `Double.MAX.VALUE` 的包装类是 java.lang.Double。

6. 当程序中需要抛出异常时，应该使用 throw 子句，当需要在方法中声明可能抛出的异常类型，应该使用 throws 子句。

7. `java.applett` 包中的 Applet 类定义了 `Applet` 与其运行环境之间的一个标准接口。

8. Java 语言中提供的 GUI 功能主要由 `java.awt` 包和 javax.swing 包中的类和接口实现。

## 三、判断题

1. Java 是 SUN 公司的产品，它是一种强数据类型语言。（对）

2. 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`。（错）

3.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。(对)

4.在定义数组时不能够直接进行内存分配,如: char ch[10]:是错误的定义语句(对)

5.拥有 abstract 方法的类是抽象类,但抽象类中可以没有 abstract,方法。(对)

四、程序阅读题,请写出程序运行结果。

1. 以下程序段的输出结果为

```
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();
 }
}
```

参考答案: 5 6 7 8 9

2. 请写出下面代码输出的结果

```
import java.io.*;
public class ATest{
 public static void main(String args[]){
 SubClass sb = new SubClass();
 System.out.println(sb.fun());
 }
}
class SuperClass{
 int a =24, b=5;
}
class SubClass extends SuperClass{
 int fun(){return a%b;}
}
```



参考答案： 4

3.

```
public class TryCatchFinally{
 static void Proc(int sel) {
 try{
 if(sel==0){
 System.out.println("no Exception ");
 return;
 }
 else if(sel==1){
 int i=0;
 int j=4/i;
 }
 }
 catch(ArithmeticException e){
 System.out.println("Catch ");}
 catch(Exception e){
 System.out.println("Will not be executed");}
 finally{
 System.out.println("finally");}
 }
 public static void main(String args[]){
 Proc(0);
 Proc(1);
 }
}
```

参考答案：

no Exception

finally

Catch

finally

## 五、写代码

1. 编写一个输出"Hello World!"的程序,用两种方式实现(Application、Applet)。(1)Application;

(2) Applet。

参考：

a.

```
public class Helloof
public static void main(string args[])
(System.out.printin("Hello World!");}
```

b.

```

import java.awt.Graphics;
import java.applet.Applet;
public class HelloApplet extends Applet{
public void paint(Graphics g)
(g.drawString("Hello World!",20,20));}

```

2. 构造方法 编写 Java 程序，模拟简单的计算器。 定义名为 **Number** 的类，其中有两个整型数据成员 **n1** 和 **n2**，应声明为私有。编写构造方法，赋予 **n1** 和 **n2** 初始值，再为该类定义加（**addition**）、减（**subtration**）、乘（**multiplication**）、除（**division**）等公有成员方法，分别对两个成员变量执行加、减、乘、除的运算。 在 **main** 方法中创建 **Number** 类的对象，调用各个方法，并显示计算结果。

```

public class Number {
 private int n1,n2;

 public Number(int n1,int n2){
 this.n1 = n1;
 this.n2 = n2 ;
 System.out.println("n1 = " + n1+ "\t" + "n2 = " + n2);
 }
// 加法
 public void addition(){
 System.out.println("n1 + n2 =" +(n1 + n2));
 }
// 减法
 public void subtration(){
 System.out.println("n1 - n2 =" +(n1 - n2));
 }
// 乘法
 public void multiplication(){
 System.out.println("n1 * n2 =" +(n1 * n2));
 }
// 除法
 public void division(){
 System.out.println("n1 / n2 =" +(n1 / n2));
 }
}

public class TestNumber {

 public static void main(String[] args) {
 Number n = new Number(10,2);

 n.addition();
 n.subtration();
 }
}

```

```

 n.multiplication();
 n.division();
 }
}

```

3. 构造方法的重载： 定义一个名为 **Vehicles**（交通工具）的基类，该类中应包含 **String** 类型的成员属性 **brand**（商标）和 **color**（颜色），还应包含成员方法 **run**（行驶，在控制台显示“我已经开动了”）和 **showInfo**（显示信息，在控制台显示商标和颜色），并编写构造方法初始化其成员属性。编写 **Car**（小汽车）类继承于 **Vehicles** 类，增加 **int** 型成员属性 **seats**（座位），还应增加成员方法 **showCar**（在控制台显示小汽车的信息），并编写构造方法。编写 **Truck**（卡车）类继承于 **Vehicles** 类，增加 **float** 型成员属性 **load**（载重），还应增加成员方法 **showTruck**（在控制台显示卡车的信息），并编写构造方法。在 **main** 方法中测试以上各类。

```

public class Vehicles {
 public String brand;//商标
 public String color;//颜色

 // 构造方法
 public Vehicles(String brand,String color){
 this.brand = brand;
 this.color = color;
 }
 // 显示信息
 public void showInfo(){
 System.out.println("车辆的商标为: "+brand+"\t"+ "颜色: "+ color);
 }
 // 行驶
 public void run(){
 System.out.println("我已经开动了! ");
 }
}

public class Car extends Vehicles {
 public int seats;

 // 构造方法
 public Car(String brand,String color,int seats){
 super(brand , color);
 this.seats = seats;
 }
 public void showCar(){
 System.out.println("小汽车商标为: "+ brand + "\t" + "颜色为: "+color + "\t" + "

```

```

座位为: "+ seats);
 }
}

public class Truck extends Vehicles{
 private float load;//载重

 // 构造方法
 public Truck(String brand,String color,float load){
 super(brand,color);
 this.load = load;
 }
 // 显示信息
 public void showTruck(){
 System.out.println("卡车的商标为: " + brand + "\t" + "颜色为: " + color + "\t" + "
载重为: " + load + "吨");
 }
}

public class TestVehicles {
 public static void main(String[] args){
 Vehicles vehicle = new Vehicles("宝马","黑色");
 vehicle.showInfo();
 Car c = new Car("玛莎拉蒂","黑色",5);
 c.showCar();

 Truck t = new Truck("奔驰", "白色", 10);
 t.showTruck();
 }
}

```

## 六、简答题

1、面向对象编程（OOP）有哪些优点？

代码开发模块化，更易维护和修改。

代码复用。

增强代码的可靠性和灵活性。

增加代码的可理解性。

## 2、面向对象编程有哪些特性？

封装、继承、多态、抽象

### 封装

封装给对象提供了隐藏内部特性和行为的能力。对象提供一些能被其他对象访问的方法来改变它内部的数据。在 Java 当中，有 3 种修饰符：**public**，**private** 和 **protected**。每一种修饰符给其他的位于同一个包或者不同包下的对象赋予了不同的访问权限。

下面列出了使用封装的好处：

通过隐藏对象的属性来保护对象内部的状态。

提高了代码的可用性和可维护性，因为对象的行为可以被单独的改变或者是扩展。

禁止对象之间的不良交互提高模块化。

### 继承

继承给对象提供了从基类获取字段和方法的能力。继承提供了代码的重用，也可以在不修改类的情况下给现存的类添加新特性。

### 多态

多态是编程语言给不同的底层数据类型做相同的接口展示的一种能力。一个多态类型上的操作可以应用到其他类型的值上面。

### 抽象

抽象是把想法从具体的实例中分离出来的步骤，因此，要根据他们的功能而不是实现细节来创建类。Java 支持创建只暴露接口而不包含方法实现的抽象的类。这种抽象技术的主要目的是把类的行为和实现细节分离开。

## 3、什么是 Java 虚拟机？为什么 Java 被称作是“平台无关的编程语言”？

Java 虚拟机是一个可以执行 Java 字节码的虚拟机进程。Java 源文件被编译成能被 Java 虚拟机执行的字节码文件。

Java 被设计成允许应用程序可以运行在任意的平台，而不需要程序员为每一个平台单独重写或者是重新编译。Java 虚拟机让这个变为可能，因为它知道底层硬件平台的指令长度和其他特性。

## 4、JDK 和 JRE 的区别是什么？

JRE(Java 运行时环境) 是将要执行 Java 程序的 Java 虚拟机。它同时也包含了执行 applet 需要的浏览器插件。JDK(Java 开发工具包) 是完整的 Java 软件开发包，包含了 JRE，编译器和其他的工具(比如：JavaDoc，Java 调试器)，可以让开发者 开发、编译、执行 Java 应用程序。

5、“static”关键字是什么意思？Java 中是否可以覆盖(override) 一个 private 或者是 static 的方法？

“static”关键字表明一个成员变量或者是成员方法可以在没有所属的类的实例的情况下被访问。

Java 中 static 方法不能被覆盖，因为方法覆盖是基于运行时动态绑定的，而 static 方法是编译时静态绑定的。static 方法跟类的任何实例都不相关，所以概念上不适用。

6、是否可以在 static 环境中访问非 static 变量？

不可以。static 变量在 Java 中是属于类的，它在所有的实例中的值是一样的。当类被 Java 虚拟机载入的时候，会对 static 变量进行初始化。如果你的代码尝试不用实例来访问非 static 的变量，编译器会报错，因为这些变量还没有被创建出来，还没有跟任何实例关联上。

7、Java 支持的数据类型有哪些？什么是自动拆装箱？

Java 支持的基本数据类型有：

byte  
short  
int  
long  
float  
double  
boolean  
char

自动装箱是 Java 编译器在基本数据类型和对应的对象包装类型之间做的一个转化。比如：把 int 转化成 Integer。反之就是自动拆箱。

8、Java 中的方法覆盖(Overriding)和方法重载(Overloading)是什么意思？

方法覆盖是说子类重新实现父类的方法。方法覆盖必须有相同的方法名，参数列表和返回类型。

方法重载发生在同一个类里面，两个或者是多个方法的方法名相同但是参数列表不同。

9、Java 中，什么是构造函数？什么是构造函数重载？什么是复制构造函数？

当新对象被创建的时候，构造函数会被调用。每一个类都有构造函数。在程序员没有给类提供构造函数的情况下，Java 编译器会为此类创建一个默认的构造函数。

Java 中构造函数重载和方法重载很相似。可以为一个类创建多个构造函数。每一个构造函数必须有它自己唯一的参数列表。

Java 不支持像 C++ 那样的复制构造函数，这个不同点是因为如果你不自己写构造函数的情况下，Java 不会创建默认的复制构造函数。

## 10、Java 支持多继承么？

不支持，Java 不支持多继承。每个类都只能继承一个类，但是可以实现多个接口。

## 11、抽象类和接口的区别是什么？

Java 支持创建抽象类和接口。它们的区别在于：

接口中所有的方法隐含的都是抽象的。而抽象类则可以同时包含抽象和非抽象的方法。

类可以实现很多个接口，但是只能继承一个抽象类

类如果要实现一个接口，它必须要实现接口声明的所有方法。但是，类可以不实现抽象类声明的所有方法，在这种情况下，类也必须得声明成是抽象的。

抽象类在实现接口时，可以不实现接口里面的方法。

Java 接口中声明的变量默认都是 `final` 的。抽象类可以包含非 `final` 的变量。

Java 接口中的成员方法默认是 `public` 的。抽象类的成员方法可以是 `private`，`protected` 或者是 `public`。

接口是绝对抽象的，不可以被实例化。抽象类也不可以被实例化，但是，如果它包含 `main` 方法的话是可以被调用的。

## 12、什么是值传递？什么是引用传递？

对象被值传递，意味着传递了对象的一个副本。因此，就算是改变了对象副本，也不会影响源对象的值。

对象被引用传递，意味着传递的并不是实际的对象，而是对象的引用。因此，外部对引用对象所做的改变会反映到所有的对象上。

## 13、进程和线程的区别是什么？

进程是执行着的应用程序，而线程是进程内部的一个执行序列。一个进程可以有多个线程。

14、创建线程有几种不同的方式？你喜欢哪一种？为什么？

创建线程有以下几种方式：

继承 `Thread` 类

实现 `Runnable` 接口

应用程序可以使用 `Executor` 框架来创建线程池

实现 `Runnable` 接口这种方式更受欢迎，因为这不需要继承 `Thread` 类。在已经继承了别的类的情况下，这需要多继承（而 `Java` 不支持多继承），只能实现接口。同时，线程池也是非常高效的，很容易实现和使用。

15、解释一下线程的几种可用状态

线程可以处于以下几种状态：

就绪(`Runnable`): 线程准备运行，不一定立马就能开始执行。

运行中(`Running`): 程序正在执行线程的代码。

等待中(`Waiting`): 线程处于阻塞的状态，等待外部的处理结束。

睡眠中(`Sleeping`): 线程被强制睡眠。

I/O 阻塞(`Blocked on I/O`): 等待 I/O 操作完成。

同步阻塞(`Blocked on Synchronization`): 等待获取锁。

死亡(`Dead`): 线程完成了执行。

16、同步方法和同步代码块的区别是什么？

同步方法就是在方法前加关键字 `synchronized`，然后被同步的方法一次只能有一个线程进入，其他线程等待。

而同步代码块则是在方法内部使用大括号使得一个代码块得到同步。同步块会有一个锁定的“对象”。同步代码块的同步范围更加准确。

17、在监视器(`Monitor`)内部，是如何做线程同步的？程序应该做哪种级别的同步？



监视器和锁在 Java 虚拟机中是一起使用的。监视器监视同步代码块，确保一次只有一个线程执行同步代码块。每一个监视器都和一个对象引用相关联。线程在获取锁之前不允许执行同步代码。

#### 18、什么是死锁(deadlock)?

两个线程都在等待对方执行完毕才能继续往下执行的时候就发生了死锁。结果就是两个线程都陷入了无限的等待中。

#### 19、如何确保 N 个线程可以访问 N 个资源同时又不导致死锁?

使用多线程的时候，一种非常简单的避免死锁的方式就是：指定获取锁的顺序，并强制线程按照指定的顺序获取锁。因此，如果所有的线程都是以同样的顺序加锁和释放锁，就不会出现死锁了。

#### 20、Java 集合类框架的基本接口有哪些?

Java 集合类里面最基本的接口有：

**Collection**：代表一组对象，每一个对象都是它的子元素。

**Set**：不包含重复元素的 **Collection**。

**List**：有顺序的 **Collection**，并且可以包含重复元素。

**Map**：可以把键(key)映射到值(value)的对象，键不能重复。

#### 21、为什么集合类没有实现 Cloneable 和 Serializable 接口?

克隆(cloning)或者是序列化(serialization)的语义和含义是跟具体的实现相关的。因此，应该由集合类的具体实现来决定如何被克隆或者是序列化。

#### 22、什么是迭代器(iterator)?

**Iterator** 接口提供了很多对集合元素进行迭代的方法。每一个集合类都包含了可以返回迭代器实例的迭代方法。迭代器可以在迭代的过程中删除底层集合的元素。

#### 23、Iterator 和 ListIterator 的区别是什么?

他们的区别如下：

**Iterator** 可用来遍历 **Set** 和 **List** 集合，但是 **ListIterator** 只能用来遍历 **List**。

Iterator 对集合只能是前向遍历，ListIterator 既可以前向也可以后向。

ListIterator 实现了 Iterator 接口，并包含其他的功能，比如：增加元素，替换元素，获取前一个和后一个元素的索引，等等。

24、快速失败(fail-fast)和安全失败(fail-safe)的区别是什么？

Iterator 的安全失败是基于对底层集合做拷贝，因此，它不受源集合上修改的影响。java.util 包下面的所有的集合类都是快速失败的，而 java.util.concurrent 包下面的所有的类都是安全失败的。快速失败的迭代器会抛出 ConcurrentModificationException 异常，而安全失败的迭代器永远不会抛出这样的异常。

25、Java 中的 HashMap 的工作原理是什么？

Java 中的 HashMap 是以键值对(key-value)的形式存储元素的。HashMap 需要一个 hash 函数，它使用 hashCode()和 equals()方法来向集合添加元素和从集合检索元素。当调用 put()方法的时候，HashMap 会计算 key 的 hash 值，然后把键值对存储在集合中合适的索引上。如果 key 已经存在了，value 会被更新成新值。

26、hashCode()和 equals()方法的重要性体现在什么地方？

Java 中的 HashMap 使用 hashCode()和 equals()方法来确定键值对的索引，当根据键获取值的时候也会用到这两个方法。如果没有正确的实现这两个方法，两个不同的键可能会有相同的 hash 值，因此，可能会被集合认为是相等的。而且，这两个方法也用来发现重复元素。所以这两个方法的实现对 HashMap 的精确性和正确性是至关重要的。

27、HashMap 和 Hashtable 有什么区别？

HashMap 和 Hashtable 都实现了 Map 接口，他们有以下不同点：

HashMap 允许键和值是 null，而 Hashtable 不允许键或者值是 null。

Hashtable 是同步的，而 HashMap 不是。因此，HashMap 更适合于单线程环境，而 Hashtable 适合于多线程环境。

HashMap 提供了可供应用迭代的键的集合，因此，HashMap 是快速失败的。另一方面，Hashtable 提供了对键的枚举(Enumeration)。

28、数组(Array)和列表(ArrayList)有什么区别？什么时候应该使用 Array 而不是 ArrayList？

Array 和 ArrayList 有以下的不同点：

Array 可以包含基本类型和对象类型，ArrayList 只能包含对象类型。

Array 大小是固定的，ArrayList 的大小是动态变化的。

ArrayList 提供了更多的方法和特性，比如：addAll(), removeAll(), iterator()等等。

对于基本类型数据，ArrayList 使用自动装箱来减少编码工作量。但是，当处理固定大小的基本数据类型的时候，这种方式相对比较慢，这时候应该使用 Array。

29、ArrayList 和 LinkedList 有什么区别？

ArrayList 和 LinkedList 有以下的不同点：

ArrayList 是基于索引的数据接口，它的底层是数组。它可以以  $O(1)$  时间复杂度对元素进行随机访问。而 LinkedList 是以链表的形式存储它的数据，每一个元素都和它的前一个和后一个元素链接在一起，在这种情况下，查找某个元素的时间复杂度是  $O(n)$ 。

相对于 ArrayList，LinkedList 的插入、添加、删除操作速度更快，因为当元素被添加到集合任意位置的时候，不需要像数组那样重新计算大小或者是更新索引。

LinkedList 比 ArrayList 更占内存，因为 LinkedList 为每一个节点存储了两个引用，一个指向前一个元素，一个指向下一个元素。

30、Comparable 和 Comparator 接口是干什么的？列出它们的区别。

Java 提供了只包含一个 compareTo() 方法的 Comparable 接口。这个方法可以个给两个对象排序。具体来说，它返回负数，0，正数来表明输入对象小于，等于，大于已经存在的对象。

Java 提供了包含 compare() 和 equals() 两个方法的 Comparator 接口。compare() 方法用来给两个输入参数排序，返回负数，0，正数表明第一个参数是小于，等于，大于第二个参数。equals() 方法需要一个对象作为参数，它用来决定输入参数是否和 Comparator 相等。只有当输入参数也是一个 Comparator 并且输入参数和当前 Comparator 的排序结果是相同的时候，这个方法才返回 true。

31、什么是 Java 优先级队列(Priority Queue)？

PriorityQueue 是一个基于优先级堆的无界队列，它的元素是按照自然顺序(natural order)排序的。在创建的时候，我们可以给它提供一个负责给元素排序的比较器。PriorityQueue 不允许 null 值，因为他们没有自然顺序，或者说他们没有任何的相关联的比较器。最后，PriorityQueue 不是线程安全的，入队和出队的时间复杂度是  $O(\log(n))$ 。

32、你了解大 O 符号(big-O notation)么？你能给出不同数据结构的例子么？

大 O 符号描述了当数据结构里面的元素增加的时候，算法的规模或者是性能在最坏的场景下有多么好。

大 O 符号也可用来描述其他的行为，比如：内存消耗。因为集合类实际上是数据结构，我们一般使用大 O 符号基于时间，内存和性能来选择最好的实现。大 O 符号可以对大量数据的性能给出一个很好的说明。

33、如何权衡是使用无序的数组还是有序的数组？

有序数组最大的好处在于查找的时间复杂度是  $O(\log n)$ ，而无序数组是  $O(n)$ 。有序数组的缺点是插入操作的时间复杂度是  $O(n)$ ，因为值大的元素需要往后移动来给新元素腾位置。相反，无序数组的插入时间复杂度是常量  $O(1)$ 。

34、Java 集合类框架的最佳实践有哪些？

根据应用的需要正确选择要使用的集合的类型对性能非常重要，比如：元素的大小是固定的，而且能事先知道，我们就应该用 `Array` 而不是 `ArrayList`。

有些集合类允许指定初始容量。因此，如果我们能估计出存储的元素的数目，我们可以设置初始容量来避免重新计算 hash 值或者是扩容。

为了类型安全，可读性和健壮性的原因总是要使用泛型。同时，使用泛型还可以避免运行时的 `ClassCastException`。

使用 JDK 提供的不变类(`immutable class`)作为 `Map` 的键可以避免为我们自己的类实现 `hashCode()`和 `equals()`方法。

编程的时候接口优于实现。

底层的集合实际上是空的情况下，返回长度是 0 的集合或者是数组，不要返回 `null`。

35、Enumeration 接口和 Iterator 接口的区别有哪些？

Enumeration 速度是 Iterator 的 2 倍，同时占用更少的内存。但是，Iterator 远远比 Enumeration 安全，因为其他线程不能够修改正在被 iterator 遍历的集合里面的对象。同时，Iterator 允许调用者删除底层集合里面的元素，这对 Enumeration 来说是不可能的。

36、HashSet 和 TreeSet 有什么区别？

HashSet 是由一个 hash 表来实现的，因此，它的元素是无序的。`add()`，`remove()`，`contains()`方法的时间复杂度是  $O(1)$ 。

TreeSet 是由一个树形的结构来实现的，它里面的元素是有序的。因此，`add()`，`remove()`，`contains()`方法的时间复杂度是  $O(\log n)$ 。

37、Java 中垃圾回收 (GC) 有什么目的？什么时候进行垃圾回收？

垃圾回收（GC）的目的是识别并且丢弃应用不再使用的对象来释放和重用资源。

38、`System.gc()`和 `Runtime.gc()`会做什么事情？

这两个方法用来提示 JVM 要进行垃圾回收。但是，立即开始还是延迟进行垃圾回收是取决于 JVM 的。

39、`finalize()`方法什么时候被调用？析构函数(`finalization`)的目的是什么？

在释放对象占用的内存之前，垃圾收集器会调用对象的 `finalize()`方法。一般建议在该方法中释放对象持有的资源。

40、如果对象的引用被置为 `null`，垃圾收集器是否会立即释放对象占用的内存？

不会，在下一个垃圾回收周期中，这个对象将是可被回收的。

41、Java 堆的结构是什么样子的？什么是堆中的永久代(`Perm Gen space`)？

JVM 的堆是运行时数据区，所有类的实例和数组都是在堆上分配内存。它在 JVM 启动的时候被创建。对象所占的堆内存是由自动内存管理系统也就是垃圾收集器回收。

堆内存是由存活和死亡的对象组成的。存活的对象是应用可以访问的，不会被垃圾回收。死亡的对象是应用不可访问尚且还没有被垃圾收集器回收掉的对象。一直到垃圾收集器把这些对象回收掉之前，他们会一直占据堆内存空间。

42、串行(`serial`)收集器和吞吐量(`throughput`)收集器的区别是什么？

吞吐量收集器使用并行版本的新生代垃圾收集器，它用于中等规模和大规模数据的应用程序。而串行收集器对大多数的小应用(在现代处理器上需要大概 100M 左右的内存)就足够了。

43、在 Java 中，对象什么时候可以被垃圾回收？

当对象对当前使用这个对象的应用程序变得不可触及的时候，这个对象就可以被回收了。

44、JVM 的永久代中会发生垃圾回收么？

垃圾回收不会发生在永久代，如果永久代满了或者是超过了临界值，会触发完全垃圾回收 (`Full GC`)。如果你仔细查看垃圾收集器的输出信息，就会发现永久代也是被回收的。这就是为什么正确的永久代大小对避免 `Full GC` 是非常重要的原因。

45、Java 中的两种异常类型是什么？他们有什么区别？

Java 中有两种异常：受检查的(`checked`)异常和不受检查的(`unchecked`)异常。不受检查的异常不需要在方法或者是构造函数上声明，就算方法或者是构造函数的执行可能会抛出这样的异

常。而且不受检查的异常可以传播到方法或者是构造函数的外面。相反，受检查的异常必须要用 `throws` 语句在方法或者是构造函数上声明。

46、Java 中 `Exception` 和 `Error` 有什么区别？

`Exception` 和 `Error` 都是 `Throwable` 的子类。`Exception` 用于用户程序可以捕获的异常情况。`Error` 定义了不期望被用户程序捕获的异常。

---

47、Java 为什么是高效的 ( High Performance )？

因为 Java 使用 `Just-In-Time` (即时) 编译器。

把 Java 字节码转换成可以直接发送给处理器的指令。

48、列举出 2 个 IDE

Eclipse

IntelliJ IDEA

49、面向对象的特征有哪些？

封装

继承

多态

抽象

50、JDK JRE JVM 分别是什么？

JDK

Java Development Kit 用作开发, 包含了 JRE, 编译器和其他的工具(比如: JavaDoc, Java 调试器), 可以让开发者开发、编译、执行 Java 应用程序。

JRE

Java 运行时环境, 是将要执行 Java 程序的 Java 虚拟机, 可以想象成它是一个容器, JVM 是它的内容。

JRE = JVM + Java Packages Classes(like util, math, lang, awt, swing etc) + runtime libraries.

## JVM

Java virtual machine (Java 虚拟机) 是一个可以执行 Java 编译产生的 Java class 文件 (bytecode) 的虚拟机进程，是一个纯的运行环境。

### 51、什么是对象 (Object)?

对象是程序运行时的实体  
它的状态存储在 fields (也就是变量)  
行为是通过方法 (method) 实现的  
方法上操作对象的内部的状态  
方法是对象对对象的通信的主要手段

### 52、一个类是由哪些变量构成的?

本地变量：在方法体，构造体内部定义的变量，在方法结束的时候就被摧毁

实例变量：在类里但是不在方法里，在类被载入的时候被实例化

类变量：在类里但是不在方法里，加了 **static** 关键字，也可以叫做静态变量

### 53、静态变量和实例变量的区别?

在语法定义上的区别：静态变量前要加 **static** 关键字，而实例变量前则不加。

在程序运行时的区别：

实例变量属于某个对象的属性，必须创建了实例对象，才能使用这个实例变量。

静态变量不属于某个实例对象，而是属于类，所以也称为类变量，只要程序加载了类的字节码，不用创建任何实例对象，静态变量就可以被使用了。

实例变量必须创建对象后才可以通过这个对象来使用，静态变量则可以直接使用类名来使用。