**软件工程之判断无敌**

1.试题标题： 单重继承 正确

所属章节： 第四章 继承与多态

所属知识点： 继承与多态 - 派生类的定义

在Java程序中，通过类的定义只能实现单重继承

2. 试题标题： 构造方法 错误

第四章 继承与多态

所属知识点： 继承与多态 - 派生类的定义

一个类中，只能拥有一个构造方法

3. 试题标题： 容器 正确

第七章 图形用户界面

所属知识点： 图形用户界面 - Swing容器

在Swing用户界面的程序设计中，容器可以被添加到其它容器中去

4. 试题标题： 数组 错误

第五章 数组、字符串与向量

所属知识点： 数组、字符串、向量 - 一维数组

定义完数组后，我们可以根据实际需要，再动态调整数组大小

5. 试题标题： 接口 正确

所属章节： 第四章 继承与多态

所属知识点： 继承与多态 - 接口

接口中的所有方法都没有被实现

6. 试题标题： 内部类

所属章节： 第三章 面向对象 错误

所属知识点： 类和对象 - 内部类和匿名类

内部类都是非静态的

7. 试题标题： 私有成员

所属章节： 第四章 继承与多态 正确

所属知识点： 类和对象 - 访问权限

类A和类B位于同一个包中，则除了私有成员，类A可以访问类B的所有其他成员

8. 试题标题： 接口名

所属章节： 第四章 继承与多态 正确

所属知识点： 继承与多态 - 接口

接口名可以作为方法返回值的类型

9. 试题标题： return语句 ：

所属章节： 第三章 面向对象 错误

所属课次： - 所属知识点： 类和对象 - 类的定义

一个方法最多能有一个return语句

10. 试题标题： 抽象类

所属章节： 第四章 继承与多态 正确

所属知识点： 继承与多态 - 抽象类

抽象类不能实例化

11. 试题标题： 运行的结果

所属章节： 第五章 数组、字符串与向量

所属知识点： 数组、字符串、向量 - 一维数组 正确

Int[] a={1,2,3,4}; System.out.println(a.length()); 以上语句运行的结果是显示4

12. 试题标题： 运行的结果

所属章节： 第三章 面向对象

所属知识点： 类和对象 - 实例方法和类方法 错误

Float aFloat=new Float(2.3); Double aDouble=new Double(2.3); if(aFloat==aDouble) System.out.println("Equal"); else System.out.println("Not Equal"); 以上语句运行的结果是显示Equal

13. 试题标题： 运行的结果 正确

if("Hunan".indexOf('n')==2) System.out.println("true"); 以上语句运行的结果是显示true

14. 试题标题： 运行的结果 错误

Character x=new Character('a'); Character y=new Character('a'); if(x.compareTo(y)) System.out.println("true"); 以上语句运行的结果是显示true

15. 试题标题： 运行的结果 错误

String str; System.out.println(str.length()); 以上语句运行的结果是显示0

16. 试题标题： 运行的结果

所属章节： 第二章 Java应用基础

所属知识点： JAVA应用基础 - 运算符 错误

int x=9; if(x>8 and x<10) System.out.println("true"); 以上语句运行的结果是显示true

17. 试题标题： 运行结果 错误

float x=26f; int y=26; int z=x/y; 以上语句能正常编译和运行

18. 试题标题： 程序的结果 错误

if(true){int i=17;}

System.out.println(i);

运行此程序的结果是输出17

19. 试题标题： 编译语句 正确

编译语句int aInt=66666不会出现编译错误

20. 试题标题： 编译语句 正确

编译语句Double aDouble=37.2D不会出现编译错误

21. 试题标题： Char 正确

试题内容Java语言中一个Char类型的数据占用2个字节大小的内存空间

22. 试题标题： 小程序 错误

一个Applet编译后的类名是Test.class，运行此小程序的命令是Java Test

23. 试题标题： 数据类型 正确

数组、类和接口都是引用数据类型

24. 试题标题： 字节码 正确

用Javac编译Java源文件后得到代码叫字节码。

25. 试题标题： 线程 错误

线程使用sleep()方法休眠后，可以用notify()方法唤醒。

26. 试题标题： private 错误

类体中private修饰的变量在本类中能访问，类生成的对象也能访问。

27. 试题标题： 包 错误

语句import java.applet.Applet；最后的Applet是代表类(class)。

28. 试题标题： 异常 正确

一个异常处理中 finally语句块只能有一个或者可以没有。

29. 试题标题： 异常 错误

程序中抛出异常时(throw …)，只能抛出自己定义的异常对象。

30. 试题标题： 输入/输出流 正确

所有的文件输入/输出流都继承于InputStream类/OutputStream类。

31. 试题标题： 接口 错误

实现一个接口，则在类中一定要实现接口中的所有方法。

32. 试题标题： System类 错误

System类中的in是一个输入对象。

33. 试题标题： 对象 错误

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。

34. 试题标题： 异常处理 错误

java异常处理中可以使用多个catch子句，此时应将高级别异常类的catch子句放在前面。

35. 试题标题： 数组 正确

Java语言中的数组元素下标总是从0开始，下标可以是整数或整型表达式。

36试题标题： 注释 错误

注释的作用是使程序在执行时在屏幕上显示//之后的内容。

37. 试题标题： 继承性 错误

由继承性可知，程序中子类拥有的成员数目一定大于等于父类拥有的成员数目。

38. 试题标题： Java语言特性 正确

Java语言采用面向对象的思想编程，具有跨平台、分布式、多线程等优点。 （ )

**问答题：**

1、

|  |
| --- |
| 当塑型前和塑型后的类中都提供了相同的方法，并将此方法发送给塑型后的对象，系统将会调用哪一个类中的方法？ |

试题答案

|  |
| --- |
| 答：对实例方法，总是从对象创建时所属的类开始，沿类层次向上查找。  对类方法，查找在编译时进行，所以总是在引用变量声明时所属的类中进行查找。 |

2、

|  |
| --- |
| 简述JDBC的功能？ |

试题答案

|  |
| --- |
| JDBC是为java语言定义的一个SQL调用级接口。JDBC使开发人员能够用纯java API来编写数据库应用程序。JDBC隔离了java与不同数据库之间的对话，使得java程序无须考虑不同的数据库管理系统平台。 |

3、

|  |
| --- |
| 简述synchronized的功能？ |

试题答案

|  |
| --- |
| 首先判断对象的锁旗标是否在，如果在就获得锁旗标，然后就可以执行紧随其后的代码段；如果对象的锁旗标不在，就进入等待状态，直到获得锁旗标。 |

4、

|  |
| --- |
| 简述java程序编译执行过程？ |

试题答案

|  |
| --- |
| 答：首先把源文件编译成字节码文件，即类文件；然后由解释器负责解释执行类文件。 |

5、

|  |
| --- |
| 试述哈希表存储对象的方式与数组、Vector及ArrayList不同？ |

试题答案

|  |
| --- |
| 答：  1） 数组，Vector及ArrayList中对象的存储位置是随机的，即对象本身与其存储位置之间没有必然的联系。因此查找一个对象时，只能以某种顺序（如顺序查找，二分查找）与各个元素进行比较，如果数组或向量中的元素数量很庞大时，查找的效率必然降低  2）哈希表中，对象的存储位置和对象的关键属性k之间有一个特定的对应关系f，我们称之为哈希(Hash)函数。它使每个对象与一个唯一的存储位置相对应。因而在查找时，只要根据待查对象的关键属性k，计算f(k)的值即可知其存储位置 |

6、

|  |
| --- |
| 抽象基类与接口的异同点？ |

试题答案

|  |
| --- |
| 答：  1）相同点  a都可以被继承。  b本身都不能被实例化  2）不同点  a抽象类是一个不完整的类，需要进一步细化；而接口只是一个行为上的规范。  b抽象基类不能实现多继承，而接口可以实现多继承。  c抽象基类可以定义字段、属性、包含有实现的方法。接口可以定义字段、索引器、事件但不能定义字段和包含实现的方法。 |

7、

|  |
| --- |
| 试述哈希表存储对象的方式与数组、Vector及ArrayList不同？ |

试题答案

|  |
| --- |
| 1）数组，Vector及ArrayList中对象的存储位置是随机的，即对象本身与其存储位置之间没有必然的联系。因此查找一个对象时，只能以某种顺序（如顺序查找，二分查找）与各个元素进行比较，如果数组或向量中的元素数量很庞大时，查找的效率必然降低  2）哈希表中，对象的存储位置和对象的关键属性k之间有一个特定的对应关系f，我们称之为哈希(Hash)函数。它使每个对象与一个唯一的存储位置相对应。因而在查找时，只要根据待查对象的关键属性k，计算f(k)的值即可知其存储位置 |

8、

|  |
| --- |
| 抽象基类与接口的异同点？ |

试题答案

|  |
| --- |
| 答：1）相同点  a都可以被继承。  b本身都不能被实例化  2）不同点  a抽象类是一个不完整的类，需要进一步细化；而接口只是一个行为上的规范。  b抽象基类不能实现多继承，而接口可以实现多继承。  c抽象基类可以定义字段、属性、包含有实现的方法。接口可以定义字段、索引器、事件但不能定义字段和包含实现的方法。 |

9、

|  |
| --- |
| 什么是线程的生命期？它包括哪几种状态？ |
|  |

试题答案

|  |
| --- |
| 线程的生命期是指从线程被创建开始到死亡的过程，通常包括5种状态：新建、就绪、运行、阻塞、死亡。 |

10、什么是继承？继承的意义？如何定义继承关系？

试题答案答：继承是一种由已有的类创建新类的机制。

通过继承可以实现代码的复用，使程序的复杂性线性地增长，而不是随规模增大呈几何级数增长。

由于父类代表了所有子类的共性，而子类既可继承其父类的共性，又可以具有本身独特的个性，在定义子类时，只要定义它本身所特有的属性与方法就可以了。

|  |
| --- |
| 5．阅读以下程序，输出结果为 。  import java.io.\*;  public class ATest{  public static void main(String args[]) {  SubClass sb = new SubClass( );  System.out.println(sb.funOfMod( )); }  }  class SuperClass{  int a = -10 , b =-3 ;  }  class SubClass extends SuperClass{  int funOfMod( ) { return a%b; }  } |

试题答案

|  |
| --- |
| -1 |

|  |
| --- |
| 4．阅读以下程序，输出结果为 。  import java.util.\*;  public class VecApp {  public static void main(String[] args) {  Vector v = new Vector();  for(int i = 0; i <2; i++)  v.addElement("v"+i);  v.insertElementAt(new Integer(1),0);  Enumeration e = v.elements();  while(e.hasMoreElements())  System.out.print(e.nextElement()+" ");  System.out.println();  v.removeElement("v1" );  for(int i = 0; i < v.size() ; i++)  System.out.print(v.elementAt(i) +" ");  System.out.println(); }  } |

试题答案

|  |
| --- |
| 1 v0 v1  1 v0 |

|  |
| --- |
| 3．以下程序段的输出结果为 。  public class C  { public static void main(String args[ ]){  int i , j ;  int a[ ] = { 2,1,4,8,9,5,3};  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( ); }  } |

试题答案

|  |
| --- |
| 1 2 3 4 5 8 9 |

|  |
| --- |
| 2．阅读以下程序：  import java.io.\*;  public class ReadLineTest{  public static void main(String[ ] args){  BufferedReader b=new BufferedReader (new InputStreamReader(System.in));  String s;  System.out.flush();  s=b.readLine();  System.out.println(s);  }  }  运行以上程序，若从键盘输入： Hello<回车>  则输出结果为 。 |

试题答案

|  |
| --- |
| Hello |

|  |
| --- |
| 1．阅读以下程序：  class A  { public static void main(String[] args)  {String s,s1="";  char c;  s=args[0];  for (int i=0;i<s.length();i++)  { c=s.charAt(i);  if(c>='a' && c<='z'){  s1=s1+Character.toUpperCase(c);  }else{ s1=s1+Character.toLowerCase(c); }  }  System.out.println(s1); }  }  若在dos命令行输入：java A hELLO，则输出为 。 |

试题答案

|  |
| --- |
| Hello |

|  |
| --- |
| import java.util.\*;  class Tree  { int height;  Tree(){ System.out.println ("Planting a seed"); height = 0; }  Tree(int i) {  height = i;  System.out.println ("Creating new Tree that is "+ height + " feet tall");  }  }  public class Overloading  { public static void main(String[] args)  {  for(int i = 1; i < 3; i++)  {  Tree t = new Tree(i);  }  new Tree();  }  } |

试题答案

|  |
| --- |
| Creating new Tree that is 1 feet tall  Creating new Tree that is 2 feet tall  Planting a seed |

|  |
| --- |
| import java.util.Vector;  import java.util.Iterator;  public class IteratorTester {  public static void main(String args[]) {  String[] num = {"china","japan","english",american"};  Vector aVector = new Vector(java.util.Arrays.asList(num));  Iterator nums = aVector.iterator();  while(nums.hasNext()) {  String aString = (String)nums.next();  System.out.println(aString);  if (aString.length() > 6) nums.remove();  }  System.out.println("After Vector: " + aVector);  }  } |

试题答案

|  |
| --- |
| china  japan |

|  |
| --- |
| public class Arrays {  public static void main(String[] args) {  int[] a1 = { 1, 2, 3, 4, 5 };  int[] a2;  a2 = a1;  for(int i = 0; i < a2.length; i++)  a2[i]++;  for(int i = 0; i < a1.length; i++)  System.out.println( "a1[" + i + "] = " + a1[i]);  }  } |

试题答案

|  |
| --- |
| a1[0]=2  a1[1]=3  a1[2]=4  a1[3]=5  a1[4]=6 |

|  |
| --- |
| class Meal {  Meal() { System.out.println("Meal()"); }  }  class Bread {  Bread() { System.out.println("Bread()"); }  }  class Lunch extends Meal {  Lunch() {System.out.println("Lunch()");}  }  class PortableLunch extends Lunch {  PortableLunch() { System.out.println("PortableLunch()"); }  }  public class Sandwich extends PortableLunch {  Bread b = new Bread();  Sandwich(){System.out.println("Sandwich()");}  public static void main(String[] args) { new Sandwich(); }  } |

试题答案

|  |
| --- |
| Meal()  Lunch()  PortableLunch()  Bread()  Sandwich() |

|  |
| --- |
| abstract class Glyph {  abstract void draw();  Glyph() {  System.out.println("Glyph() before draw()");  draw();  System.out.println("Glyph() after draw()");  }  }  class RoundGlyph extends Glyph {  int radius = 1;  RoundGlyph(int r) {  radius = r;  System.out.println("RoundGlyph.RoundGlyph(), radius = " + radius);  }  void draw() {  System.out.println("RoundGlyph.draw(), radius = " + radius);  }  }  public class PolyConstructors {  public static void main(String[] args) {  new RoundGlyph(5);  }  } |

试题答案

|  |
| --- |
| Glyph() before draw()  RoundGlyph.draw(), radius = 0  Glyph() after draw()  RoundGlyph.RoundGlyph(), radius = 5 |

|  |
| --- |
| class A1{  int x = 20;  public void setx(int i){  x = i;  }  void printa(){  System.out.println(x);  }  }  class B1 extends A1{  int x=1;  void printb() {  super.x = super.x +10 ;  System.out.println  ("super.x= " + super.x +  " x= " + x);  }  }  public class Exam4\_4Test  {  public static void main(String[] args){  A1 a1 = new A1();  a1.setx(4);  a1.printa();  B1 b1 = new B1();  b1.printb();  b1.printa();  b1.setx(6);  b1.printb();  b1.printa();  a1.printa();  }  } |

试题答案

|  |
| --- |
| 1、 4  super.x= 30 x= 1  30  super.x= 16 x= 1  16  4 |

|  |
| --- |
| public class BankAccount {  private int accountNumber;  private float balance;  public BankAccount(int number, float initBal){  accountNumber = number;  balance = initBal;  }  public String toString(){  return("Account #"+new java.text.DecimalFormat("000000").format(accountNumber) + " with balance " + new java.text.DecimalFormat("$0.00").format(balance));  }  }  public class AccountTester {  public static void main(String args[]) {  BankAccount anAccount;  anAccount = new BankAccount(100023,100);  System.out.println(anAccount);  }  } |

试题答案

|  |
| --- |
| Account #100023 with balance $100.00 |

|  |
| --- |
| public class Test {  public static void main(String[] args) {  String s1="I like cat";  StringBuffer sb1=new StringBuffer ("It is Java");  String s2;  StringBuffer sb2;  s2=s1.replaceAll("cat","dog");  sb2=sb1.delete(2,4);  System.out.println("s1为："+s1);  System.out.println("s2为："+s2);  System.out.println("sb1为："+s1);  System.out.println("sb2为："+s2);  }  }  运行结果： |

试题答案

|  |
| --- |
| s1为：I like cat  s2为：I like dog  sb1为：I like cat  sb2为：I like dog |

|  |
| --- |
| import java.util.\*;  public class Hello {  public static void main(String[] args) {  String s="Friday;Saturday\\Sunday Monday,Tuesday";  StringTokenizer stk=new StringTokenizer(s,"; \\");  while(stk.hasMoreTokens()){  System.out.println(stk.nextToken());  }  }  }  运行结果： |

试题答案

|  |
| --- |
| Friday  Saturday  Sunday  Monday,Tuesday |

|  |
| --- |
| class Daemon extends Thread {  public void run() {  if(this.isDaemon()==false)  System.out.println("thread is not daemon");  else  System.out.println("thread is daemon");  try {  for(int i=0;i<10;i++){  System.out.println(i);  Thread.sleep(200);  }  }catch (InterruptedException e){}  System.out.println("thread done!");  }  }  public class Test {  public static void main (String[] args) {  Thread t=new Daemon();  t.setDaemon(true);  t.start();  try {  Thread.sleep(900);  }catch (InterruptedException e){}  System.out.println("main done!");  }  }  运行结果： |

试题答案

|  |
| --- |
| thread is daemon  0  1  2  3  4  main done! |

|  |
| --- |
| public class Arrays {  public static void main(String[] args) {  int[] a1 = { 1, 2, 3 };  int[] a2;  a2 = a1;  for(int i = 0; i < a2.length; i++)  a2[i]=a2[i]+1;  for(int i = 0; i < a1.length; i++)  System.out.println( "a1[" + i + "] = " + a1[i]);  }  }  运行结果： |

试题答案

|  |
| --- |
| a1[0]=2  a1[1]=3  a1[2]=4 |

|  |
| --- |
| class Test{  public static void main(String args[]) {  int a=25, b=20, e=3, f=0;  boolean d=a<b;  System.out.println(d);  if (e!=0 && a/e>5)  System.out.println(a/e);  if (f!=0 && a/f>5)  System.out.println(a/f);  else  System.out.println(f);  }  }  运行结果： |

试题答案

|  |
| --- |
| false  8  0 |

|  |
| --- |
| class Ball{  public double radius;  public Ball(){}  public Ball(double r){  radius=r; }  }  class MovingBall extends Ball{  private double speed;  public MovingBall(double s){  speed=s;  }  void show(){  System.out.println(super.radius);  System.out.println(speed);  }  }  public class Test{  public static void main(String args[]){  MovingBall mb=new MovingBall(20);  mb.show();  }  }  运行结果： |

试题答案

|  |
| --- |
| 0.0  20.0 |

|  |
| --- |
| class A1{  int x = 20;  public void setx(int i){  x = i;}  void printa(){  System.out.println(x);}  }  class B1 extends A1{  int x=1;  void printb() {  super.x = super.x +10 ;  System.out.println("super.x= " + super.x + " x= " + x);}  }  public class Exam4\_4Test {  public static void main(String[] args){  A1 a1 = new A1();  a1.setx(4);  a1.printa();  B1 b1 = new B1();  b1.printb();  b1.printa();  b1.setx(6);  b1.printb();  b1.printa();}  } |

试题答案

|  |
| --- |
| 1、4  super.x= 30 x= 1  30  super.x= 16 x= 1  16 |

|  |
| --- |
| public class Arrays {  public static void main(String[] args) {  int[] a1 = { 1, 2, 3, 4, 5 };  int[] a2;  a2 = a1;  for(int i = 0; i < a2.length; i++)  a2[i]=a2[i]+i;  for(int i = 0; i < a1.length; i++)  System.out.println( "a1[" + i + "] = " + a1[i]);  }  } |

试题答案

|  |
| --- |
| 1、a1[0]=1  a1[1]=3  a1[2]=5  a1[3]=7  a1[4]=9 |

|  |
| --- |
| public class ArrayCopyDemo {  public static void main(String[] args) {  char[] copyFrom = {'d','e','c','a','f','f','e','i','n','a'};  char[] copyTo = new char[7];  System.arraycopy(copyFrom, 2, copyTo, 0, 7);  System.out.println(new String(copyTo));  }  } |

试题答案

|  |
| --- |
| caffein |

|  |
| --- |
| public class BankAccount {  private int accountNumber;  private float balance;  public BankAccount(int number, float initBal){  accountNumber = number;  balance = initBal;}  public String toString(){  return("Account #" + accountNumber +" with balance " + new java.text.DecimalFormat("$0.00").format(balance));}}  public class AccountTester {  public static void main(String args[]) {  BankAccount anAccount;  anAccount = new BankAccount(100023,100);  System.out.println(anAccount);}  } |

试题答案

|  |
| --- |
| Account #100023 with balance $100.00 |

|  |
| --- |
| abstract class Glyph {  abstract void draw();  Glyph() {  System.out.println("Glyph() before draw()");  draw();  System.out.println("Glyph() after draw()");}  }  class RoundGlyph extends Glyph {  int radius = 1;  RoundGlyph(int r) {  radius = r;  System.out.println("RoundGlyph(),radius="+ radius);}  void draw() {  System.out.println("draw(), radius = " + radius);}}  public class PolyConstructors {  public static void main(String[] args) {new RoundGlyph(5);}  } |

试题答案

|  |
| --- |
| Glyph() before draw()  draw(), radius = 0  Glyph() after draw()  RoundGlyph(), radius = 5 |

|  |
| --- |
| class Meal {  Meal() { System.out.println("Meal()"); }  }  class Bread {  Bread() { System.out.println("Bread()"); }  }  class Lunch extends Meal {  Lunch() {System.out.println("Lunch()");}  }  class PortableLunch extends Lunch {  PortableLunch() { System.out.println("PortableLunch()"); }  }  public class Sandwich extends PortableLunch {  Bread b = new Bread();  Sandwich(){System.out.println("Sandwich()");}  public static void main(String[] args) {  new Sandwich(); }  } |

试题答案

|  |
| --- |
| Meal()  Lunch()  PortableLunch()  Bread()  Sandwich() |

|  |
| --- |
| public class Arrays  { public static void main(String[] args)  { int[] a1 = { 1, 2, 3, 4, 5 };  int[] a2;  a2 = a1;  for(int i = 0; i < a2.length; i++) a2[i]+i;  for(int i = 0; i < a1.length; i++)  System.out.println( "a1[" + i + "] = " + a1[i]);  }  } |

试题答案

|  |
| --- |
| a1[0]=1  a1[1]=3  a1[2]=5  a1[3]=7  a1[4]=9 |

|  |
| --- |
| public class ArrayCopyDemo  { public static void main(String[] args)  { char[] copyFrom = { 'd', 'e', 'c', 'a', 'f', 'f', 'e',  'i', 'n', 'a', 't', 'e', 'd'};  char[] copyTo = new char[7];  System.arraycopy(copyFrom, 2, copyTo, 0, 7);  System.out.println(new String(copyTo));  }  } |

试题答案

|  |
| --- |
| caffein |

**编程题**

|  |
| --- |
| 2． 编写程序，完成文件复制功能，即将a.txt文件内容复制到b.txt文件中去。 |

试题答案

|  |
| --- |
| //Hello.html  <html>  <applet code=Hello height=400 width=200>  </applet>  </applet>  2．  import java.io.\*;  public class CopyFile  {  public static void main(String args[]) throws Exception  {  String s, space=" ";  int num=0;  BufferedReader br = new BufferedReader(new FileReader("a.txt"));  BufferedWriter bw = new BufferedWriter(new FileWriter("b.txt"));  while (br.ready())  {  s=br.readLine();  if (s==null) break;    bw.write(s);  bw.newLine();  }  bw.close();  }  } |

**2、**

|  |
| --- |
| 1．编写一个输出"Hello World！"的Applet程序和嵌入该Applet的HTML页面。 |

试题答案

|  |
| --- |
| 1．  ①//Hello.java  import java.awt.Graphics;  　import java.applet.Applet;  　public class Hello extends Applet{  　　String s;  　public void init(){ s=" Hello World!";　}  　public void paint(Graphics g){ g.drawString(s,25,25);　}  } |

**3、**

|  |
| --- |
| 编写一个完整的Java Application 程序。包含类Student、TestStudent。  具体要求如下：  (1)Student类 ：  ①属性: （5’）  name : String对象，表示一个人姓名  sex： char类型，用来表示性别  id：long类型，表示学号  classinfo：String对象，表示班级  address ：String对象，表示家庭地址  ②方法: （20’）  Student (String name, char sex, long id)：构造函数  String getName() ：返回姓名  void setId(long id) ：设置学号  void setAddress(String add) ：设置家庭地址  void setClass(String classinfo) ：设置班级信息  public String toString() ：返回学生的各项信息，包括姓名、性别等上述属性  (2) TestStudent类作为主类要完成测试功能（15’）  ①用以下信息生成一个Student对象aGirl。  姓名： 邓亚萍  性别： 女  学号： 1234567  ②设置家庭地址: 江西农业大学42号  设置班级信息：2004计算机1班  ③输出对象aGirl的各项信息 |

试题答案

|  |
| --- |
| 4、class Student  {  protected String name;  protected char sex;  protected long id;  protected String classinfo;  protected String address;  Student(String name, char sex, long id)  { this.name = name;  this.sex = sex;  this.id = id;  }  void setId(long id )  { this.id=id;  }  void setAddress(String addr)  { this.address = addr;  }  void setClass(String classinfo)  { this.classinfo =classinfo;  }    public String toString()  { String s = new String( "姓名: " + name+ "性别: " + sex);  s += "学号: " + id;  if (classinfo != null) s += "班级: " + classinfo;  if (address != null) s += "家庭住址: " +address;  return s;  }  }    public class TestStudent  {  public static void main(String args[])  { Student aGirl = new Student("邓亚萍", '女',1234567);  aGirl.setClass("2004计算机1班");  aGirl.setAddress("江西农业大学42号");  System.out.println("aGirl: " + aGirl);  }  } |

**4、**

|  |
| --- |
| 从键盘上输入一行，以空格为分隔符将此行分成多个单独的子字符串，并按每行一个子字符串的格式输出。 |

试题答案

|  |
| --- |
| 3、import java.io.\*;  import java.util.StringTokenizer;  public class ex  {  public static void main(String[ ] args)throws IOException{  BufferedReader in = new BufferedReader(  new InputStreamReader(System.in));  String s="",s1;  while((s1=in.readLine()).length()!=0)  s+=s1;  StringTokenizer st=new StringTokenizer(s);  while (st.hasMoreTokens())  System.out.println(st.nextToken());  }  } |

**5、**

|  |
| --- |
| 设计一个类，引用第2题设计的包。计算长为6cm，宽为8cm，高为10cm的长方体的体积。 |

试题答案

|  |
| --- |
| import cn.edu.jxau.soft.class06x.\*;  public class PackageTest    {  public static void main(String[ ] args)  {  A MyobjA=new A();  System.out.println("column is:"B.FunB(6,8,10));  }} |

**6、**

|  |
| --- |
| 设计一个类，引用第2题设计的包。计算长为6cm，宽为8cm，高为10cm的长方体的体积。 |

试题答案

|  |
| --- |
| import cn.edu.jxau.soft.class06x.\*;  public class PackageTest    {  public static void main(String[ ] args)  {  A MyobjA=new A();  System.out.println("column is:"B.FunB(6,8,10));  }} |

**7、**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 设有一ACCESS数据库的ODBC数据源为：source，用户名为：tester，密码为：1234。  数据库中表person的数据如下：   |  |  | | --- | --- | | ID | name | | 1 | john | | 2 | hl |   请用JDBC-ODBC桥来连接数据库，输出数据库中所有数据，并插入一条记录（3，”dh”）到数据库中。 |

试题答案

|  |
| --- |
| import java.sql.\*;  public class ex10\_1 {  public static void main(String[] args) throws Exception {  String DBDriver="sun.jdbc.odbc.JdbcOdbcDriver";  String connectionStr="jdbc:odbc:source";  Connection con = null;  Statement stmt = null;  ResultSet rs = null;  Class.forName(DBDriver);  con=DriverManager.getConnection(connectionStr,"testert","1234");  stmt=con.createStatement();  rs=stmt.executeQuery("Select \* From Person");  while(rs.next()) {  System.out.print(rs.getInt("ID")+" ");  System.out.println(rs.getString("Name")+" ");  };  stmt.executeUpdate("INSERT INTO Person VALUES(3,'dh')");  stmt.close(); //关闭语句  con.close(); //关闭连接  }} |

**8、**

|  |
| --- |
| 设计一线程随机产生一整数放入一存储器中，存储器一次只能存放一个数。另一线程将打印出此整数。要求两线程要协同工作，先存后打印此数。共循环10次。 |

试题答案

|  |
| --- |
| public class aa{  public static void main(String[] args){  memory t=new memory(10);  new Producer(t).start();  new Consumer(t).start();  }  }  class memory{  int size;  int number=0;  int produce=0;  boolean available=false;  public memory(int size){  this.size=size;  }  public synchronized void put(){  if(available)  try{wait();} catch(Exception e){}  produce=(int)(Math.random() \* 10 );  System.out.println("Producer puts memory "+produce);  ++number;  available=true;  notify();  }  public synchronized void sell(){  if(!available)  try{wait();} catch(Exception e){}  System.out.println("Consumer remove memory "+produce);  available=false;  notify();  if (number==size) number=size+1 ;  }  }  class Producer extends Thread{  memory t=null;  public Producer(memory t){  this.t=t;  }  public void run(){  while(t.number<t.size)  t.put();  }  }  class Consumer extends Thread{  memory t=null;  public Consumer(memory t){  this.t=t;  }  public void run(){  while(t.number<=t.size)  t.sell();  }  } |

**9、**

|  |
| --- |
| 使用Vector存储四个字符串china、japan、english、american。然后将长度大于6的字符串从中移除。 |

试题答案

|  |
| --- |
| import java.util.Vector;  import java.util.Iterator;  public class IteratorTester {  public static void main(String args[]) {  String[] num = {"china","japan","english",american"};  Vector aVector = new Vector(java.util.Arrays.asList(num));  Iterator nums = aVector.iterator();  while(nums.hasNext()) {  String aString = (String)nums.next();  System.out.println(aString);  if (aString.length() > 6) nums.remove();  }  System.out.println("After Vector: " + aVector);  }  } |

**10、**

|  |
| --- |
| 请自定义一个包Package1，包中含两个类，一个类用来计算正方形面积，另一个类用来计算长方形体积。在类test当中使用Package1包中的类计算一边长为10正方形面积，计算一长方形(10×20×30)的体积。 |

试题答案

|  |
| --- |
| package Package1;  public class A  {  public void FunA(int x)  {  Return x\*x;  }  }  class B  {  public void FunB(int x, int y, int z)  {  Return x\*y\*z;  }  }  /\*\*\*\*test\*\*\*\*\*\*\*\*\*/  import PackageA.\*;  public class PackageTest  {  public static void main(String[ ] args)  {  A MyobjA=new A();  B MyobjB=new B();  System.out.println("area is:"A.FunA(10));  System.out.println("column is:"B.FunB(10,20,30));  }} |

**11、**

|  |
| --- |
| 从屏幕上输入一个整数，然后输出它的翻转形式。 |

试题答案

|  |
| --- |
| import java.io.\*;  public class ex3\_5  {  public static void main(String[ ] args)throws IOException{  int n, right\_digit, newnum = 0;  BufferedReader in = new BufferedReader(  new InputStreamReader(System.in));  System.out.println("Enter the number: ");  n=(new Integer(in.readLine())).intValue();  System.out.print("The number in reverse order is ");  do {  right\_digit = n % 10;  System.out.print(right\_digit);  n /= 10;  }  while (n != 0);  System.out.println();  }  } |

**12、**

|  |
| --- |
| 设计一线程随机产生一整数放入一存储器中，存储器一次只能存放一个数。另一线程将打印出此整数。要求两线程要协同工作，先存后打印此数。共循环10次。 |

试题答案

|  |
| --- |
| public class aa{  public static void main(String[] args){  memory t=new memory(10);  new Producer(t).start();  new Consumer(t).start();  }  }  class memory{  int size;  int number=0;  int produce=0;  boolean available=false;  public memory(int size){  this.size=size;  }  public synchronized void put(){  if(available)  try{wait();} catch(Exception e){}  produce=(int)(Math.random() \* 10 );  System.out.println("Producer puts memory "+produce);  ++number;  available=true;  notify();  }  public synchronized void sell(){  if(!available)  try{wait();} catch(Exception e){}  System.out.println("Consumer remove memory "+produce);  available=false;  notify();  if (number==size) number=size+1 ;  }  }  class Producer extends Thread{  memory t=null;  public Producer(memory t){  this.t=t;  }  public void run(){  while(t.number<t.size)  t.put();  }  }  class Consumer extends Thread{  memory t=null;  public Consumer(memory t){  this.t=t;  }  public void run(){  while(t.number<=t.size)  t.sell();  }  } |

**13、**

|  |
| --- |
| 使用Vector存储四个字符串china、japan、english、american。然后将长度大于6的字符串从中移除。 |

试题答案

|  |
| --- |
| import java.util.Vector;  import java.util.Iterator;  public class IteratorTester {  public static void main(String args[]) {  String[] num = {"china","japan","english",american"};  Vector aVector = new Vector(java.util.Arrays.asList(num));  Iterator nums = aVector.iterator();  while(nums.hasNext()) {  String aString = (String)nums.next();  System.out.println(aString);  if (aString.length() > 6) nums.remove();  }  System.out.println("After Vector: " + aVector);  }  } |

**14、**

|  |
| --- |
| 请自定义一个包Package1，包中含两个类，一个类用来计算正方形面积，另一个类用来计算长方形体积。在类test当中使用Package1包中的类计算一边长为10正方形面积，计算一长方形(10×20×30)的体积。 |

试题答案

|  |
| --- |
| package Package1;  public class A  {  public void FunA(int x)  {  Return x\*x;  }  }  class B  {  public void FunB(int x, int y, int z)  {  Return x\*y\*z;  }  }  /\*\*\*\*test\*\*\*\*\*\*\*\*\*/  import PackageA.\*;  public class PackageTest  {  public static void main(String[ ] args)  {  A MyobjA=new A();  B MyobjB=new B();  System.out.println("area is:"A.FunA(10));  System.out.println("column is:"B.FunB(10,20,30));  }} |

**15、**

|  |
| --- |
| 从屏幕上输入一个整数，然后输出它的翻转形式。 |

试题答案

|  |
| --- |
| import java.io.\*;  public class ex3\_5  {  public static void main(String[ ] args)throws IOException{  int n, right\_digit, newnum = 0;  BufferedReader in = new BufferedReader(  new InputStreamReader(System.in));  System.out.println("Enter the number: ");  n=(new Integer(in.readLine())).intValue();  System.out.print("The number in reverse order is ");  do {  right\_digit = n % 10;  System.out.print(right\_digit);  n /= 10;  }  while (n != 0);  System.out.println();  }  } |

**16、**

|  |
| --- |
| 定义一个表示学生的类student，成员变量有学号、姓名、性别、年龄，方法有获得学号、姓名、性别、年龄；修改年龄。书写Java程序创建student类的对象啊并在main方法中调用student类中的方法。 |

试题答案

|  |
| --- |
| public class student{  private int stu\_ID;  private String name;  private String sex;  private int old;  student(int id,String name,String sex,int old){  stu\_ID=id;  this.name=name;  this.sex=sex;  this.old=old;  }  void show\_id(){  System.out.println("the student ID is:"+stu\_ID);  }  void show\_name(){  System.out.println("the student name is:"+name);  }  void show\_sex(){  System.out.println("the student sex is:"+sex);  }  void show\_old(){  System.out.println("the student old is:"+old);  }  void change\_old(int newyear){  old=newyear;  }  public static void main(String args[]){  student Lee=new student(2007070130,"Li Ming","M",18);  Lee.show\_id();  Lee.show\_name();  Lee.show\_sex();  Lee.show\_old();  Lee.change\_old(20);  Lee.show\_old();  }  } |

**17、**

|  |
| --- |
| 用穷举法求出3位数中百、十、个位数的立方和就是该数的数。 |

试题答案

|  |
| --- |
| public class Test  { public static void main(String[] args)  {  int a,b,c,x=100;  while(x<1000){  a=x%10;  b=(x%100-a)/10;  c=(x-x%100)/100;  if(a\*a\*a+b\*b\*b+c\*c\*c==x)  System.out.println(x);  x+=1;  }  }  } |

**18、**

|  |
| --- |
| 题目：一球从100米高度自由落下，每次落地后反跳回原高度的一半；再落下，求它在第10次落地时，共经过多少米？第10次反弹多高？ |

试题答案

|  |
| --- |
| public class lianxi10 { public static void main(String[] args) {  double h = 100,s = 100;  for(int i=1; i<10; i++) {  s = s + h;  h = h / 2;  }  System.out.println("经过路程：" + s);  System.out.println("反弹高度：" + h / 2); } } |

**19、**

|  |
| --- |
| 一个数如果恰好等于它的因子之和，这个数就称为 "完数 "。例如6=1＋2＋3.编程 找出1000以内的所有完数。 |

试题答案

|  |
| --- |
| public class lianxi09 { public static void main(String[] args) {  System.out.println("1到1000的完数有： ");  for(int i=1; i<1000; i++) {  int t = 0;  for(int j=1; j<= i/2; j++) {  if(i % j == 0) {  t = t + j;  }  }  if(t == i) {  System.out.print(i + " ");  }  } } |

**20、**

|  |
| --- |
| 求s=a+aa+aaa+aaaa+aa...a的值，其中a是一个数字。例如2+22+222+2222+22222(此时共有5个数相加)，几个数相加有键盘控制。 |

试题答案

|  |
| --- |
| import java.util.\*; public class lianxi08 { public static void main(String[] args) {  long a , b = 0, sum = 0;  Scanner s = new Scanner(System.in);  System.out.print("输入数字a的值： ");  a = s.nextInt();  System.out.print("输入相加的项数：");  int n = s.nextInt();  int i = 0;  while(i < n) {  b = b + a;  sum = sum + b;  a = a \* 10;  ++ i;  }  System.out.println(sum); } } |

**21、**

|  |
| --- |
| 输入一行字符，分别统计出其中英文字母、空格、数字和其它字符的个数。 |

试题答案

|  |
| --- |
| import java.util.\*; public class lianxi07 { public static void main(String[] args) { int digital = 0; int character = 0; int other = 0; int blank = 0;  char[] ch = null;  Scanner sc = new Scanner(System.in);  String s = sc.nextLine();  ch = s.toCharArray();  for(int i=0; i<ch.length; i++) {  if(ch >= '0' && ch <= '9') {  digital ++;  } else if((ch >= 'a' && ch <= 'z') || ch > 'A' && ch <= 'Z') {  character ++;  } else if(ch == ' ') {  blank ++;  } else {  other ++;  }  }  System.out.println("数字个数: " + digital);  System.out.println("英文字母个数: " + character);  System.out.println("空格个数: " + blank);  System.out.println("其他字符个数:" + other );  }  } |

**22、**

|  |
| --- |
| 输入两个正整数m和n，求其最大公约数和最小公倍数。 |

试题答案

|  |
| --- |
| import java.util.\*; public class lianxi06 {  public static void main(String[] args) { int a ,b,m; Scanner s = new Scanner(System.in); System.out.print( "键入一个整数： ");  a = s.nextInt(); System.out.print( "再键入一个整数： ");  b = s.nextInt();  deff cd = new deff();  m = cd.deff(a,b);  int n = a \* b / m;  System.out.println("最大公约数: " + m);  System.out.println("最小公倍数: " + n); }  } class deff{ public int deff(int x, int y) {  int t;  if(x < y) {  t = x;  x = y;  y = t;  }   while(y != 0) {  if(x == y) return x;  else {  int k = x % y;  x = y;  y = k;  }  }  return x; } } |

**23、**

|  |
| --- |
| 利用条件运算符的嵌套来完成此题：学习成绩> =90分的同学用A表示，60-89分之间的用B表示，60分以下的用C表示。 |

试题答案

|  |
| --- |
| import java.util.\*; public class lianxi05 { public static void main(String[] args) {  int x;  char grade;  Scanner s = new Scanner(System.in);  System.out.print( "请输入一个成绩: ");   x = s.nextInt();   grade = x >= 90 ? 'A'  : x >= 60 ? 'B'  :'C';  System.out.println("等级为："+grade);   } } |

**24、**

|  |
| --- |
| 将一个正整数分解质因数。例如：输入90,打印出90=2\*3\*3\*5。 |

试题答案

|  |
| --- |
| import java.util.\*; public class lianxi04{   public static void main(String[] args) {  Scanner s = new Scanner(System.in);  System.out.print( "请键入一个正整数: ");   int n = s.nextInt();  int k=2;   System.out.print(n + "=" );  while(k <= n) {  if(k == n) {System.out.println(n);break;}  else if( n % k == 0) {System.out.print(k + "\*");n = n / k; }   else k++;  }  }  } |

**25、**

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

试题答案

|  |
| --- |
| public class lianxi03 { public static void main(String[] args) {  int b1, b2, b3;   for(int m=101; m<1000; m++) {   b3 = m / 100;  b2 = m % 100 / 10;  b1 = m % 10;  if((b3\*b3\*b3 + b2\*b2\*b2 + b1\*b1\*b1) == m) {  System.out.println(m+"是一个水仙花数"); }  } } } |

**26、**

|  |
| --- |
| 判断101-200之间有多少个素数，并输出所有素数。 |

试题答案

|  |
| --- |
| public class lianxi02 { public static void main(String[] args) {  int count = 0;  for(int i=101; i<200; i+=2) {  boolean b = false;  for(int j=2; j<=Math.sqrt(i); j++)   {  if(i % j == 0) { b = false; break; }   else { b = true; }  }  if(b == true) {count ++;System.out.println(i );}  }  System.out.println( "素数个数是: " + count); } } |

**27、**

|  |
| --- |
| 题目：古典问题：有一对兔子，从出生后第3个月起每个月都生一对兔子，小兔子长到第三个月后每个月又生一对兔子，假如兔子都不死，问每个月的兔子总数为多少？ |

试题答案

|  |
| --- |
| public class lianxi01 { public static void main(String[] args) { System.out.println("第1个月的兔子对数: 1"); System.out.println("第2个月的兔子对数: 1"); int f1 = 1, f2 = 1, f, M=24;  for(int i=3; i<=M; i++) {  f = f2;  f2 = f1 + f2;  f1 = f;  System.out.println("第" + i +"个月的兔子对数: "+f2);  } } } |

感谢小强，感谢小明，感谢铎哥为编排工作花费巨大人力，财力，物力。希望每位同学好好复习，好好迎接期末考试！