一、判断题。判断下列各题是否正确，若正确，在（）内打“√”，否则打“×”。

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。

2、Java语言的源程序不是编译型的，而是编译解释型的。

3、Java语言使用的是Unicode字符集，每个字符在内存中占8位。

4、Java语言中不同数据类型的长度是固定的，不随机器硬件不同而改变。

6、一个类可以实现多个接口，接口可以实现“多重继承”。

7、实现接口的类不能是抽象类。

二、不定项选择题。

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、main方法是Java Application程序执行的入口点，关于main方法的方法首部，以下（）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[])

3、关于下面代码的说法（）是正确的？

public class Example{

public static void main(String[] args){

int[] arr= new int[10];

}

}

A、编译时产生错误

B、编译时正确，运行时将产生错误

C、输出零

D、输出空(null)

4、不正确的表达式是（）。

A、float a=1.0d;

B、Double a=new Double(1.0)

C、byte a = (byte)340;

D、Byte a = 120;

5、定义一个类名为"MyClass.java"的类，并且该类可被一个工程中的所有类访问，那么该类的正确声明应为（ ）。

A、private class MyClass extends Object

B、class MyClass extends Object

C、public class MyClass

D、public class MyClass extends Object

6、下列构造方法的调用方式中，正确的是(）。

A、按照一般方法调用

B、由用户直接调用

C、只能通过new自动调用

D、被系统调用

7、private级的成员可以被哪些对象访问？C

A、同一包中的其他类

B、仅同一类的私有成员方法

C、仅同一类的所有成员方法

D、所有其他类

8、关于Object 类，以下正确的是（）。

A、Object类中定义了一些重要的方法，以方便Java类的使用

B、Object类是一个非常特殊的“虚拟”类，就是说只能从Object类派生出其它类，但不能创建一个Object类的对象，也不能将一个引用变量定义为Object类型

C、如果一个类已经有了父类，那么这个类同时继承了Object类，也就是说这时候这个类实际上有两个父类

D、Object类是所在其他Java 类的直接或间接父类

E、Java 中的类是单继承的，因此所在的类以Object类为根，形成了棵继承树

9、抽象类与普通类的区别是什么？

A、抽象类没有成员变量，只有成员函数

B、抽象类只有成员变量，没有成员函数

C、抽象类中有抽象函数，而普通类的每个函数都是有具体实现的

D、抽象类与普通类没有区别

10、使用接口的通常办法是（）。

A、将接口数当作静态数据来调用

B、直接调用接口的方法

C、将接口变量作为方法的参数来使用

D、定义一个接口变量，该变量指向实现接口的对象来访问

E、定义一个接口对象，通过对象访问

11、执行下列代码后，哪个结论是正确的 String[] s=new String[10];

A、s[10] 为 ""

B、s[9] 为 null;

C、s[0] 为 未定义

D、s.length 为10

12、以下对substring 函数的描述中，错误的是（）。

A、String 和StringBuffer类都可以使用substring 函数

B、substring 函数返回一个StringBuffer对象

C、substring 函数获得从指定位置开始，到指定位置为止的字符串

D、substring 函数用于获得字符串的一个子串

13、如果catch块中包括多个异常类，这些异常类的匹配原则是什么？

A、接try块中程序执行的顺序，寻找最通用的异常类

B、接try块中程序执行的顺序，寻找能够最精确匹配的类

C、接catch块中异常类的排列顺序，寻找第一个能够匹配的异常对象

D、在catch块中寻找能够最精确匹配的类

14、以下异常中，不属于RuntimeException 的是（）。

A、IndexoutofBoundsException 数组以下越界异常

B、NullPointerExcePtion 使用未初始化的引用变量

C、ArithmeticException 算术异常

D、IllegalAccessException没有足够的权限访问其他类的成员

15、运行下列程序, 会产生什么结果。

public class MyThread extends Thread implements Runnable{

　public void run(){

　}

　public static void main(String[] args) {

　　Thread t=new Thread(new MyThread ());

　　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

三、简答题。

1、子类重新定义与父类方法的方法头完全相同的方法，这种情况称为什么?此时子类对象还能执行父类中原来的方法吗?

2、final修饰符可以用来修饰什么?被final修饰符修饰后有何特点?

3、试写出一个语句创建一个名为myPackage的包，这个语句应该放在程序的什么位置?

**四、填空题：**

## 1. 根据结构组成和运行环境的不同，java程序可以分为\_\_\_\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_\_\_\_两类。

## 2. Java源程序文件编译后产生的文件称为\_\_\_\_\_\_\_\_\_文件，其拓展名为\_\_\_\_\_\_\_\_\_。

## 3. 面对对象方法中，类的实例称为\_\_\_\_\_\_\_\_\_。

## 4. Java源文件中最多只能有一个\_\_\_\_\_\_\_\_\_类，其他类的个数不限。

## 5. Java的几个版本分别是\_\_\_\_\_\_\_\_\_，\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_。

## 6. 对于代码int x，a=2，b=3，c=4；x=++a+b+++c++；的执行结果是\_\_\_\_\_\_\_。

## 7. Java语言中的浮点型数据根据储存长度和数值精度的不同，进一步分为\_\_\_\_\_\_\_和\_\_\_\_\_\_\_两种具体类型。

## 8. 当整型变量n的值不能被13除尽时，其值为false的Java语言表达式是\_\_\_\_\_\_\_。

## 9. Java的引用数据类型有类、数组和\_\_\_\_\_\_\_。

## 10. 已知：boolean b1=true，b2；则表达式：!b1&&b2||b2的值为\_\_\_\_\_\_\_\_\_\_\_\_。

## 11. 顺序结构、选择结构和\_\_\_\_\_\_\_\_是结构化程序设计的3种基本流程控制结构。

## 12. 每一个else字句都必须和一个距离它最近的\_\_\_\_\_\_\_字句相对应。

## 13. 在switch语句中，break语句的作用是：执行完一个\_\_\_\_\_\_\_\_分支后跳出\_\_\_\_\_\_\_语句。

## 14. 循环语句包括for循环、\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_。

## 15. \_\_\_\_\_\_\_语句的功能是：跳过循环体内部下面未执行的语句，回到循环体开始位置，继续下次循环。

## 16. Java中声明数组包括数组的名字，数组包含的元素的\_\_\_\_\_\_\_。

## 17. 数组声明后，必须使用\_\_\_\_\_\_运算符分配内存空间。

## 18. 在Java中，所有的数组都有length属性，这个属性存储了该数组的\_\_\_\_\_\_\_。

## 19. 声明数组仅仅是给出了数组名字和元素的数据类型，想要真正地使用数组还必须为它\_\_\_\_\_\_\_。

## 20. 设有数组定义：int a[ ] = { 1,2,3,4,5,6,7,8,9 }；则执行下列几个语句后的输出结果是\_\_\_\_\_。

## 21. 类的修饰符分为\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_。

## 22. 程序中定义类使用的关键字是\_\_\_\_\_\_\_\_\_，每个类的定义由类头定义、类体定义两部分组成，其中类体部分包括\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_。

## 23. main方法的声明格式是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

## 24. 4．创建一个类的对象的运算符是\_\_\_\_\_\_\_\_\_。

## 25. java源文件中最多只能有一个\_\_\_\_\_\_\_\_\_类，其他的类的个数不限。

## 26. 类方法不能直接访问其他所属类的\_\_\_\_\_\_\_\_\_变量和\_\_\_\_\_\_\_\_\_方法，只可直接访问其所属类的\_\_\_\_\_\_\_\_\_变量和\_\_\_\_\_\_\_\_\_方法。

## 27. 类成员的访问控制符有\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_和默认4种。

## 28. protected类型的类成员可被同一\_\_\_\_\_\_\_\_\_、同一包中的\_\_\_\_\_\_\_\_\_和不同包中的\_\_\_\_\_\_\_\_\_的代码访问引用。

## 29. 下面是一个类的定义：

public class \_\_\_\_\_\_\_\_\_{

intx,y;

Myclass(int i, \_\_\_\_\_){ //构造方法

x=I;

y=j;

}

}

## 30. Java使用固定于首行的\_\_\_\_\_\_\_\_\_\_语句来创建包。

## 31. 在运行时，由Java解释器自动引入，而不用import语句引入的包是\_\_\_\_\_\_\_\_\_\_。

## 32. 系统规定有\_\_\_\_\_\_\_\_\_\_表示当前类的构造方法，用\_\_\_\_\_\_\_\_\_\_表示直接父类的构造方法，在构造方法中两者只能选其一，且须放在第一条语句。

## 33. \_\_\_\_\_\_\_\_\_\_直接赋值给\_\_\_\_\_\_\_\_\_\_时，子类对象可自动转换为父类对象， \_\_\_\_\_\_\_\_\_\_赋值给\_\_\_\_\_\_\_\_\_\_时，必须将父类对象强制转化为子类对象。

## 34. Java语言中，定义子类对象时，使用关键字\_\_\_\_\_\_\_\_\_\_来声明父类名。如果没有指出父类，则该类的默认父类为\_\_\_\_\_\_\_\_\_\_。

## 35. 如果一个类包含一个或多个abstract方法。则它是一个\_\_\_\_\_\_\_\_\_\_类。

## 36. Java不直接支持多继承，但可以通过\_\_\_\_\_\_\_\_\_\_实现多继承。类的继承具有\_\_\_\_\_\_\_\_\_\_性。

## 37. \_\_\_\_\_\_\_\_\_\_是Java程序中所有类的直接或间接父类，也是类库中所有的父类。

## 38. 定义初值为10的10次方的长整型变量lvar的语句是\_\_\_\_\_\_\_\_\_\_

## 39. 以下方法m的功能是求两参数之积的整数部分。

int m(float x, float y){

\_\_\_\_\_\_\_\_\_\_

}

## 40. \_\_\_\_\_\_\_\_\_\_包包含了Collection的接口的类的API。

## 41. Math.round(11.5)等于\_\_\_\_\_\_\_\_\_\_，Math.round(-11.5)等于\_\_\_\_\_\_\_\_\_\_。

## 42. \_\_\_\_\_\_\_\_对象可以使用read方法从标准的输入设备（通常键盘）读取数据；\_\_\_\_\_\_\_\_\_\_对象可以使用print方法向标准输出设备（屏幕）输出显示。

## 43. 框架（JFrame）和面板（JPanel）的默认布局管理器分别是\_\_\_\_\_\_和\_\_\_\_\_\_\_。

## 44. Swing的布局管理器主要包括\_\_\_\_\_\_\_。

## 45. Java事件处理包括建立事件源、\_\_\_\_\_\_\_\_和将事件源注册到监听器 。

## 46. AWT的事件处理机制包括\_\_\_\_\_\_\_、事件和事件监听者。

## 47. Swing的顶层容器有\_\_\_\_\_\_\_\_、JApplet、JWwindow和JDialog。

## 48. 线程的启动是通过调用其\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法而实现的。

## 49. Java虚拟机(JVM)中的线程调度器负责管理线程，调度器把线程的优先级分为10个级别，分别用Thread类中的类常量表示，每个Java线程的优先级都在常数\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_之间，即Thread.MIN\_PRIORIY和Thread.MAX\_PRIORIY之间。