**Java基础知识测试**

考试时间：2小时

1. 简答题（55分）
2. 方法重载和方法重写（覆盖）的区别。（6分）

重载：方法参数不一样，在同一个类内

重写：继承或实现关系下，对父类方法的覆盖，参数任意

1. 写出java.lang.Object类的六个常用方法的声明并说明其作用。（6分）

Public void equals(Object obj) 比较引用类型变量的内容

protected void finalize() 当垃圾回收器确定不存在对该对象的更多引用时，由对象的垃圾回收器调用此方法。

String toString() 返回该对象的字符串表示

int hashCode() 返回该对象的哈希码值。

void notify() 唤醒在此对象监视器上等待的单个线程。

void notifyAll() 唤醒在此对象监视器上等待的所有线程。

1. 接口和抽象类的联系和区别。（8分）

联系：不能实例化，都需要子类继承或实现，为子类提供规范

区别：接口中只有抽象方法，抽象类可以没有抽象方法，格式不一样，抽象类中含有构造方法。

1. final、finally、finalize的区别。（6分）

final是修饰符，用于修饰类、方法、变量，表示最终的不可修改；

finally放在try catch语句后面，无论是否捕获异常，finally后面的语句都会执行；

finalize

1. Error和Exception的区别， Checked异常和Runtime异常的区别。（6分）

Error是虚拟机错误，不需要处理；exception是程序异常，分为编译时异常和运行时异常，编译时异常时必须被处理的

1. String、StringBuffer、StringBuilder区别与联系。（9分）

String类代表字符串，StringBuffer和StringBuilder都是String的子类

StringBuffer是线程安全的可变字符序列 JDK1.0开始,

StringBuilder JDK1.5开始 比StringBuffer快但线程不安全

1. Java异常处理中，关键字try、catch、finally、throw、throws分别代表什么含义？（6分）

try后面跟可能有异常出现的语句；

catch捕获不同类型的异常并做出处理；

finally在try catch之后，不管有没有捕获到异常，它后面的语句都会被执行；

throw在方法名后抛出异常，跟异常类名

throws抛出异常对象，后跟异常对象

1. 简述static关键字的作用。

提示：从static可以修饰变量，方法，代码块，内部类四个方面来回答。（8分）

修饰变量：变量在内存中只存在一份，被所有该类对象共享；

修饰方法：可以通过类名直接调用方法，静态方法不能调用非静态的方法；

代码块：第一次加载类时，静态代码块中的语句都会被执行

1. 编码题（45分）
   1. 写出饿汉式单例模式、懒汉单例模式的示例代码。（15分）

饿汉式

**public** **class** SingleTonE {

**private** **static** SingleTonE *instance* = **new** SingleTonE();

**private** SingleTonE() {}

**public** **static** SingleTonE getInstance() {

**return** *instance*;

}

}

懒汉式

**public** **class** SingleTonL {

**private** **static** SingleTonL *instance*;

**private** SingleTonL() {}

**public** **static** SingleTonL getInstance() {

**if**(*instance* == **null**) {

*instance* = **new** SingleTonL();

}

**return** *instance*;

}

}

* 1. 编写递归算法程序：一列数的规则如下: 0、1、1、2、3、5、8、13、21、34...... 求数列的第40位数是多少。（10分）

**public** **class** Recursion {

**int** resultNum;

**public** **int** getNum(**int** num) {

**if**(num == 1) {

**return** 0;

} **else** **if**(num == 2) {

**return** 1;

} **else** {

**return** getNum(num-1) + getNum(num-2);

}

}

@Test

**public** **void** test() {

System.***out***.println(getNum(40)); // 结果：63245986

}

}

* 1. 编写应用程序，创建类的对象，分别设置圆的半径、圆柱体的高，计算并分别显示圆半径、圆面积、圆周长，圆柱体的体积。（20分）

**提示:**（1）编写一个圆类Circle，该类拥有：

* + - 1. 一个成员变量，radius（私有，浮点型）;//存放圆的半径；
      2. 两个构造方法

Circle（） //将半径设为0

Circle（double r ） //创建Circle对象时将半径初始化为r

* + - 1. 三个成员方法

double getArea（） //获取圆的面积

double getPerimeter（） //获取圆的周长

void show（） //将圆的关径、周长、面积输出到屏幕

**public** **class** Circle {

**private** **double** radius;

**public** **static** **final** **double** ***PI*** = 3.14;

**public** Circle() {

radius = 0;

}

**public** Circle(**double** r) {

radius = r;

}

**public** **double** getArea() {

**return** ***PI***\*radius\*radius;

}

**public** **double** getPerimeter() {

**return** 2\****PI***\*radius;

}

**public** **void** show() {

System.***out***.println("该圆半径：" + radius + " 周长：" + getPerimeter() + " 面积：" + getArea());

}

}

测试

@Test

**public** **void** testC() {

**new** Circle(3).show();

}

输出结果 该圆半径：3.0 周长：18.84 面积：28.259999999999998

（2） 编写一个圆柱体类Cylinder，它继承于上面的Circle类。还拥有：

* + 1. 一个成员变量，double hight （私有，浮点型）; //圆柱体的高；
    2. 构造方法

//创建Cylinder对象时将半径初始化为r,高度初始化为h

Cylinder（double r,double h）

* + 1. 成员方法

double getVolume（） //获取圆柱体的体积

void showVolume（） //将圆柱体的体积输出到屏幕

**public** **class** Cylinder **extends** Circle {

**private** **double** hight;

**public** Cylinder(**double** r, **double** h) {

**super**(r);

hight = h;

}

**public** **double** getVolume() {

**return** ***PI***\*radius\*radius\*hight;

}

**public** **void** showVolume() {

System.***out***.println("该圆柱面积为：" + getVolume());

}

}