1. 选择题 (记分： 单选题 2分一题 多选题 3分一题)

1. (单选题)关于下列代码说法正确的是：

public class Money {

private String country, name;

public String getCountry() {

return country;

}

}

class Yen extends Money {

public String getCountry() {

return super.country;

}

}

class Euro extends Money {

public String getCountry(String timeZone) {

return super.getCountry();

}

}

A. Yen类编译正确

B. Euro类编译正确

C. Money类编译错误

D. Yen和Money编译错误

2.(单选题)请看下列程序的输出结果是：（ ）

public class Item {

private String desc;

public String getDescription() {

return desc;

}

public void setDescription(String d) {

desc = d;

}

public static void modifyDesc(Item item, String desc) {

item = new Item();

item.setDescription(desc);

}

public static void main(String[] args) {

Item it = new Item();

it.setDescription("Gobstopper");

Item it2 = new Item();

it2.setDescription("Fizzylifting");

modifyDesc(it, "Scrumdiddlyumptious");

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

System.out.println(it2.getDescription());

}

}

A. Scrumdiddlyumptious Scrumdiddlyumptious

B. Scrumdiddlyumptious Fizzylifltng

C. Gobstopper Scrumdiddlyumptious

D. Gobstopper Fizzylifting

3.(单选题)下列代码的运行结果是（）。

public class Animal {

public String noise() {

return "peep";

}

public static void main(String[] args) {

Animal animal = new Dog();

Cat cat = (Cat)animal;

System.out.println(cat.noise());

}

}

class Dog extends Animal {

public String noise() {

return "bark";

}

}

class Cat extends Animal {

public String noise() {

return "meow";

}

}

A. peep

B. bark

C. meow

D. 抛出运行时异常

4.(单选题)请看下列代码：

public static void main(String[] args) {

<插入代码>

System.out.println(s);

}

如果程序输出的结果是4997，那么在处应该填入代码是（）。

A. String s = "123456789"; s = (s-"123").replace(1,3,"99") - "89";

B. StringBuffer s = new StringBuffer("123456789"); s.delete(0,3).replace( 1,3, "99").delete(4,6);

C. StringBuffer s = new StringBuffer("123456789"); s.substring(3,6).delete( 1 ,3).insert( 1, "99");

D. StringBuilder s = new StringBuilder("123456789"); s.substring(3,6).delete( 1 ,2).insert( 1, "99");

5.(单选题)下面的代码用于对数组arr实现冒泡排序：

for (int i = 0; i < arr.length - 1; i++) {

boolean isSwap = false;

空白处

if (!isSwap)

break;

}

下列选项中，空白处可以填入的代码是：（）。

1. for (int j = arr.length - 1; j > i; j--) {

if (arr[j] < arr[j - 1]) { int temp = arr[j]; arr[j] = arr[j - 1]; arr[j - 1] = temp; isSwap = true; } }

B. for (int j = arr.length - 1; j > 0; j--) { if (arr[j] < arr[j - 1]) { int temp = arr[j]; arr[j] = arr[j - 1]; arr[j - 1] = temp; isSwap = true; } }

C. for (int j = i + 1; j< arr.length; j++) { if (arr[j] < arr[j - 1]) { int temp = arr[j]; arr[j] = arr[j - 1]; arr[j - 1] = temp; isSwap = true; } }

D. for (int j = i; j< arr.length; j++) { if (arr[j] < arr[j - 1]) { int temp = arr[j]; arr[j] = arr[j - 1]; arr[j - 1] = temp; isSwap = true; } }

**6.(单选题)下列代码运行的结果是（）。**

**public class Base {**

**public static final String FOO = "foo";**

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

**Base b = new Base();**

**Sub s = new Sub();**

**System.out.print(Base.FOO);**

**System.out.print(Sub.FOO);**

**System.out.print(b.FOO);**

**System.out.print(s.FOO);**

**System.out.print(((Base) s).FOO);**

**}**

**}**

**class Sub extends Base {**

**public static final String FOO = "bar";**

**}**

**A. Foofoofoofoofoo**

**B. foobarfoobarfoo**

**C. foobarfoofoofoo**

**D. foobarfoobarbar**

7. (单选题)下列表达式中，可以得到精确结果的是（）。

A. double d1 = 3.0 - 2.6;

B. double d4 = 2.5 \* 1.5;

C. double d2 = 30/300;

D. double d3 = 1/2 + 0.5;

8. (单选题)(单选题)定义如下两个变量：

int x = 89;

int y = 98;

下列选项中可以将两个变量值互换的是：（）。

A.x = y;

y = x;

B.int temp = 0；

temp = x;

y = x;

x = temp;

C.int temp = 0；

temp = x;

x = y;

y = temp;

D.int temp = 0；

temp = y;

x = y;

y = temp;

9. (单选题)关于下列代码说法正确的是：

class ClassA {

public int numberOfinstances;

protected ClassA(int numberOfinstances) {

this.numberOfinstances = numberOfinstances;

}

}

public class ExtendedA extends ClassA {

private ExtendedA(int numberOfinstances) {

super(numberOfinstances);

}

public static void main(String[] args) {

ExtendedA ext = new ExtendedA(420);

System.out.print(ext.numberOfinstances);

}

}

A.

运行后，输出420

B.

运行时抛出异常

C.

编译错误，所有的构造器必须是public的

D.

编译错误，构造器不能是private的

10.(单选题)请看下列代码：

interface Foo {

int bar();

}

public class Sprite {

public int fubar(Foo foo) {

return foo.bar();

}

public void testFoo() {

fubar(

<插入代码>

);

}

}

使类Sprite编译通过，在处应填入的代码是：

A.

Foo { public int bar() { return 1; } }

B.

new Foo { public int bar() { return 1; } }

C.

new Foo() { public int bar(){return 1; } }

D.

new class Foo { public int bar() { return 1; } }

11. (多选题)下列赋值语句中，会有编译错误的是（）。

A. int a = 8888888888;

B. char b = 1000+300;

C. byte c = 100+30;

D. int d = 'a'+'b'+'c';

12. (多选题)在Java语言中，下列说法正确的是：（）。

A.

StringBuffer和StringBuilder的区别在于：StringBuffer是线程安全的而StringBuilder不是。

B.

String是不可变对象，而StringBuffer中封装的字符串数据是可以动态改变的。

C.

判断两个StringBuilder对象的字符序列是否相同，可以调用其equlas方法进行比较。

D.

String的重写了equals方法，重写的逻辑是：字符序列相同的String对象equals方法返回true。

13. (多选题) 请看下列代码：

package com.tarena;

public class Geodetics {

public static final double DIAMETER = 12756.32;

}

访问静态常量DIAMETER的方式正确的是：

A. import com.tarena.Geodetics;

public class TerraCarta {

public double halfway(){

return Geodetics.DIAMETER/2.0;

}

}

B. import com.tarena.Geodetics;

public class TerraCarta {

public double halfway(){

return DIAMETER/2.0;

}

}

C. import com.tarena;

public class TerraCarta {

public double halfway(){

return Geodetics.DIAMETER/2.0;

}

}

D. import com.tarena.\*;

public class TerraCarta {

public double halfway(){

return Geodetics.DIAMETER/2.0;

}

}

14. (多选题)题目代码的功能为：输出每个字符在一个字符串中出现的次数(不区分大小写)。

String str = "ewrwqFrewqfsadfdsfdsfs";

str=str.toLowerCase();

int max\_length = 0;

while (str.length() > 0) {

《插入代码》

}

A.

int length = str.length(); char first=str.charAt(0); String strNew = str.replaceAll(String.valueOf(first), ""); if (length>strNew.length()) { max\_length = length - strNew.length(); System.out.println(first+":"+max\_length); }

B.

int length = str.length(); char first=str.charAt(0); String strNew = str.replaceAll(String.valueOf(first), ""); if (length>strNew.length()) { max\_length = length - strNew.length(); str = strNew; System.out.println(first+":"+max\_length); }

C.

int length = str.length(); String first = str.substring(0, 1); String strNew = str.replaceAll(first, ""); if (length>strNew.length()) { max\_length = length - strNew.length(); str = strNew; System.out.println(first+":"+max\_length); }

D.

int length = str.length(); String first = str.substring(0, 1); String strNew = str.replaceAll(first, ""); if (length>strNew.length()) { max\_length = length - strNew.length(); System.out.println(first+":"+max\_length); }

15. (多选题) 请看下列代码：

public class Old {

public static Object get(List list) {

return list.get(0);

}

}

以下选项调用get方法，能编译通过的是：

A. Object o = Old.get(new LinkedList());

B. Object o = Old.get(new LinkedList<?>());

C. String s = Old.get(new LinkedList<String>());

D. String s = (String)Old.get(new LinkedList<String>());

16.(多选题)请看下列代码：

public class Key {

private long id1;

private long 1d2;

// class Key methods

}

程序员开发Key类，作为java.util.HashMap的key，那么Key应该覆盖的两个方法是：

A.

public int hashCode()

B.

public boolean equals(Key k)

C.

public int compareTo(Object o)

D.

public boolean equals(Object o)

17.(多选题)请看下列代码：

public abstract class Shape {

int x;

int y;

public abstract void draw();

public void setAnchor(int x, int y) {

this.x = x;

this.y = y;

}

}

下列选项中能正确使用Shape类的是：

A.

public class Circle implements Shape { private int radius; }

B.

public abstract class Circle extends Shape { private int radius; }

C.

public class Circle extends Shape { private int radius; public void draw(); }

D.

public class Circle extends Shape { private int radius; public void draw() {/\* code here \*/} }

**17.(多选题)请看下列代码：**

**public class Tetromino {**

**protected int move(int x) {**

**return 0;**

**}}**

**class T extends Tetromino {**

**<插入代码>}**

**在<插入代码>处填入选项中的代码，使T类没有编译错误的是：（ ）。**

**A.public int move(int x) { return 0; }**

**B.private int move(int x) { return 0; }**

**C.private int move(long x) { return 0; }**

**D.protected long move(int x) { return 0; }**

**18.(多选题)下列关于JVM的内存结构描述正确的是：（ ）。**

**A.类的各种信息在堆中保存**

**B.栈用于存放程序运行过程当中所有的局部变量**

**C.堆一般用于存储使用new关键字创建的对象**

**D.类是JVM的内存结构**

**19.(多选题)请看下列代码：**

**public class Foo {**

**static void alpha() { /\* more code here \*/}**

**void beta() { /\* more code here \*/}}**

**下列说法正确的是：（ ）。**

**A.Foo.beta()是调用beta方法的正确方式**

**B.Foo.alpha()是调用alpha方法的正确方式**

**C.beta方法可以直接调用alpha方法**

**D.alpha方法可以直接调用beta方法**

**20.(多选题)请看下列代码：**

**public class Line {**

**private Point p;**

**public Line() {**

**<插入代码>}**

**public class Point { }**

**}**

**在<插入代码>处，需要创建Point类的对象，下列选项正确的是：（ ）。**

**A.super.p = new Point();**

**B.p = super.new Point();**

**C.p =new Point();**

**D.p =this.new Point();**

1. **简答题 (记分： 3分一题 总共：30分 )**

**(1) 我们常用的java API中常用的类 至少写出5个.**

**(2)** 说说&和&&的区别。

**（3）**"=="和equals方法究竟有什么区别？

**(4 )** String s = "Hello";s = s + " world!";这两行代码执行后，原始的String对象中的内容到底变了没有？

**(5 ) java中的基本数据类型有哪些?String是基本数据类型吗?**

**(6 )** 面向对象的特征有哪些方面?

**(7 )** Overload和Override的区别。Overloaded的方法是否可以改变返回值的类型?

**(8 )** 静态变量和实例变量的区别?

**(9 )** 一个".java"源文件中是否可以包括多个类（不是内部类）？有什么限制？

**(10 )** 接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承具体类(concrete class)? 抽象类中是否可以有静态的main方法？

**三．编程题（记分： 10分一题 总共：20分）**

1. 验证登录信息的合法性

(1) 在Eclipse中创建类文件，类文件包含包名 com.mingrisoft ，类名CheckLogin

(2) 并在该类的主方法中接收用户输入的用户登录名和登录密码，然后通过if条件语句

分别判断用户名和密码，并输出登录验证结果。

1. 在控制台显示效果是：

1 2 3 4

10 11 12 5

9 8 7 6

提示： （该现象名为矩阵）使用二维数组来进行存值，循环输出值