周测

一 选择题(每题3分,共45分)

1.(单选题)读下列程序，对类型转换操作正确的是（ ）。

B

class A{}

class B extends A{}

class C extends A{}

A o1 = new A();

A o2 = new B();

A o3 = new C();

A . A a = o1; B b = o2; C c = o3;

B. A a = o1 ; B b = (B) o2; C c = (C) o3;

C. A a = (A)o1; B b = (B)o3; C c = (C)o2;

D. A a = (A)o2; B b = (B)o1; C c = (C)o3;

2.(单选题)在Java中， 负责对字节码解释执行的是（）。

C

A.应用服务器

B.垃圾回收器

C.虚拟机

D.编译器

3. 下（）不是Object类的方法

D  
A) split(）

B) equals()

C) toString（）

D) hasNext（）

4.(单选题)下列程序段运行后，输出（ ）。

C

String str = "123az45d67xyz89";

str = str.replaceAll("[a-z]+", "\*\*\*");

System.out.println(str);

A.123az45d67xyz89

B.\*\*\*az\*\*\*d\*\*\*xyz\*\*\*

C.123\*\*\*45\*\*\*67\*\*\*89

D.123456789

5.(单选题)对于如下字符串格式“+86 13523456789”字符串中“+86”可出现也可以不出现，如果出现，只出现1次，后有任意位数空格，并有11位数字，匹配如上需求，下列正确的正则表达式是（ ）。

C

A.^\\+86\\s\*\\d{11}$

B.^(\\+86)\*\\s\*\\d{11}$

C.^(\\+86)?\\s\*\\d{11}$

D.^(\\+86)+\\s\*\\d{11}$

6.(单选题)下列程序运行结果是（）。

C

public class Test {

public static void main(String args[]){

String str=new String("tedu");

char[]ch={'x','y','z'};

Test t = new Test();

t.change(str,ch);

System.out.print(t.str+"and");

System.out.print(t.ch);}

public void change(String str,char ch[]){

str="tarena";

ch[0]='w'; }

}

A.tedu and xyz

B.tarena and xyz

C.tedu and wyz

D.tarena and wyz

7.(单选)关于String 和 StringBuilder 下面说法正确的是（）。

A

A.String操作字符串不改变原有字符串的内容

B. StringBuilder连接字符串速度没有String快

C.String可以使用append方法连接字符串

D. StringBuilder在java.util包中

8.(单选)请看下列代码

B

public class Member{

private Long userId;

private String nickName;

//以下是getter和sett方法 ……

}

main方法中的代码：

Member m1=new Member();

m1.setUserId(new Long(100001));

m1.setNickName("mick");

Member m2=new Member();

m2.setUserId(new Long(100001));

m2.setNickName("mick");

System.out.println(m1==m2);

System.out.println(m1.equals(m2));

控制台的输出结果是：()

A.true false

B.false true

C.false false//s未重写 与==相同

D.true ture

9.(单选题)阅读下面程序段:

A 刷新c

BufferedOutputStream bos = new BufferedOutputStream( new FileOutputStream("bos.txt"));

bos.write( "hello".getBytes(“utf-8”)));

假设bos.txt不存在，运行该程序段后，文件bos.txt的字节数为:（）

A.5 B.10 C.4 D.0

10.(单选题)阅读下列程序代码段：

C

try {

RandomAccessFile raf = new RandomAccessFile("c:/a.txt", "rw");

raf.write('A');

raf.writeInt(300);

System.out.println(raf.getFilePointer());

} catch (FileNotFoundException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace(); }

该程序输出的结果是（ ）。

A.1

B.2

C.5

D.4

11.(单选题)请看下列代码：

D

public class Foo {

public static void main (String[]args) throws Exception {

String name = "tarena";//6

byte[] c = name.getBytes();

try {

File f = new File("company.txt")；

FileOutputStream out = new FileOutputStream(f, true);

out.write(c);

out.close();

}catch (IOException e) {}

}

}

假设文件company.txt中已存在一些内容，则关于上述代码的说法正确的是：（）。

A.编译错误

B.程序可以通过编译，运行后文件company.txt没有任何改变

C.程序可以通过编译，运行后文件company.txt的长度变为0

D.程序可以通过编译，运行后company.txt文件末尾增加了字符串“tarena”

12.(单选题)请看下列代码：

B

public static void main(String[] args) throws Exception{

FileOutputStream fos = new FileOutputStream("record.dat");

BufferedOutputStream bos= new BufferedOutputStream(fos);

bos.write('a');

}

关于上述代码编译运行的结果的是：（）。

A.编译错误

B.实现向文件record.dat追加写入字符a

C.实现向文件record.dat覆盖写入字符a

D.文件record.dat无任何内容

13.(单选题)假设有random.dat文件，其内容为“International”，请看下列代码的输出结果是：（ ）。

B

RandomAccessFile raf = new RandomAccessFile("random.dat", "r");

raf.seek (5);//n

byte[] buf = new byte[8];

raf.read(buf);

System.out.println(new String(buf, “utf-8”));

raf.close();

A.International

B.national

C.rnational

D.ational

14.(单选题)下列代码中，能实现按行读取文件的数据的流是：（）。

B

A.InputStreamReader

B.BufferedReader

C.FileInputStream

D.InputStream

15.(单选题)下列代码运行的结果是：（）。

B

public class Boxer {

Integer i;

int x;

public Boxer(int y) {

x = i + y;

System.out.println(x);

}

public static void main(String[] args) {

new Boxer(new Integer(4));

}

}

A.输出：4

B.在运行时抛出NullPointerException异常

C.在运行时抛出NumberFormatException异常

D.在运行时抛出IllegalStateException异常

二 简答题:(每题10分,共20分)

1.String str=new String("abc")创建了几个对象?在上述创建过对象后的基础上在创建

String str1=new String("abc")一共创建了几个对象?并说明原因

第一个String str = new String(“abc”)；中创建了两个对象，”abc“因为字符串堆中没有，因此在字符串池中创建第一个对象；new String()会创建一个新的对象，同时接受了”abc“参数，代表它的值是之前就已经创建的”abc”.最后str是String类型变量接受对象的应用赋值。

第二个String str1 = new String(“abc”);中创建了一个对象，”abc“因为字符串对中有，不会创建新的；new String();会创建一个新的，接受参数。并将引用赋值给str1。

2.==和equest在做比较时的区别?

==：如果是基本类型会比较两边的值，如果是引用类型会比较两边的地址值。

Equal：比较结果取决于equal方法的具体实现，String，Integer等都重写了equals()方法，比较的是数值或者内容，Object则是比较对象是否相等。

三、程序题：（共35分）

按照题目完成一下操作：

1. 创建Person类，类中的内容有：（10分）
   1. 私有属性 ： 姓名，年龄，性别，人员描述（desc）
   2. 无参构造方法，全参数构造方法
   3. 每个属性的set，get方法
   4. 重写toString（）方法，返回值为各属性拼接成的字符串，格式为： “tom,22,男，是一个帅气的人！”
2. 创建测试类，完成以下操作：（25分）
   1. 用户从控制台输入一个人的信息（以字符串的形式），格式为: “tom,22,男，是一个帅气的人！”
   2. 将接受到的用户信息解析出来并封装成Person对象，然后通过ObjectOuputStream对象将该person对象写入person.obj文件，并测试，看是否写入
      1. 注意：desc属性的值不需要写入文件
   3. 然后完成从person.obj文件中读取信息并还原成Person对象，最后将Person对象的信息打印输出。

package cn.tedu.serializable**;**import com.sun.org.apache.xerces.internal.impl.xs.SchemaNamespaceSupport**;**import java.io.\***;**import java.util.Scanner**;**public class Test {  
 public static void main(String[] args) {  
 System.*out*.println("输入一个人的信息！")**;** Scanner input = new Scanner(System.*in*)**;** String in = input.nextLine()**;** Person p=new Person()**;** p.setName(in.split("\\,")[**0**])**;** p.setAge(Integer.*parseInt*(in.split("\\,")[**1**]))**;** p.setGender(in.split("\\,")[**2**])**;** p.setDesc(in.split("\\,")[**3**])**;** System.*out*.println(p.toString())**;** System.*out*.println("开始写入")**;** //序列化  
 ObjectOutputStream out =null**;** try {  
 out = new ObjectOutputStream(new FileOutputStream("person.obj"))**;** out.writeObject(p)**;**// out.writeObject(p.getName());  
// out.writeInt(p.getAge());  
// out.writeUTF(p.getGender());  
 }catch (IOException e){  
 e.printStackTrace()**;** }finally {  
 try {  
 out.close()**;** System.*out*.println("成功写入")**;** } catch (IOException e) {  
 e.printStackTrace()**;** }  
 }  
 ObjectInputStream ois = null**;** System.*out*.println("开始读入")**;** try {  
 ois = new ObjectInputStream(new FileInputStream("person.obj"))**;** Object o=ois.readObject()**;** System.*out*.println(o.toString())**;** }catch (Exception e){  
 e.printStackTrace()**;** }finally {  
 try {  
 ois.close()**;** } catch (IOException e) {  
 e.printStackTrace()**;** }  
 }  
  
 }  
}  
  
class Person implements Serializable {  
 private static final long *serialVersionUID*=**1L;** private String name**;** private int age**;** private String gender**;** private String desc**;** public String getName() {  
 return name**;** }  
  
 public void setName(String name) {  
 this.name = name**;** }  
  
 public int getAge() {  
 return age**;** }  
  
 public void setAge(int age) {  
 this.age = age**;** }  
  
 public String getGender() {  
 return gender**;** }  
  
 public void setGender(String gender) {  
 this.gender = gender**;** }  
  
 public String getDesc() {  
 return desc**;** }  
  
 public void setDesc(String desc) {  
 this.desc = desc**;** }  
  
 @Override  
 public String toString() {  
 return name + "," + age + "," + gender + "," + desc**;** }  
}