

江西师范大学

考试卷

年级 14 级 专业(学科)网络工程、计算机科学 15 —16 学年 2 学期

课程号: 262194 课程名称: 面向对象程序设计 ☒ A 卷 ☐ 卷

考试形式: ☐ 卷 ☒ 闭卷 其他要求: 请发草稿纸

一、基础知识题 (每小题 10 分, 共计 30 分)

1、类 X 将输出 n 层的数字三角 (第 i 行打印 i 个 i)。请将其补充完成。

注: 用打印语句罗列输出结果将不给分。

```
class X{
    public static void main (String[] args) { int i,j;
        /* 请补充完整, 使之能得到期望的结果 */
    }
}
```

输出结果为:

```
1
2 2
3 3 3
```

2、下面程序将交换两个对象的数据。请补充完成类 Data 和 App, 使其能得到给定的输出结果。注: 用打印语句罗列输出结果将不给分。

```
class Data{ private int d;
    public Data(int x){d=x;}
}
class App{
    public static void main (String[] args) {
        Data d1=new Data(3); Data d2=new Data(5);
        switchData(d1,d2); //交换两个对象
        showResult(d1,d2); //依次打印 d1、d2 对象中的数据    }
}
```

输出结果为:

```
5 3
```

3、下列代码用线程演示 3 个线程的同步: 按 t1-t2-t3 这一特定次序依次输出。每个线程每轮只能输出一个数组元素。

```
class MySignal{
    private int data=0;
```

```
public void setData(int x, int y){ //当 data 值为 x 时运行, 之后将其改为 y
```

```
    /* 请补充完整, 使之能得到期望的结果 */
```

```
    }
}
class T implements Runnable{
    private Thread t;    private MySignal flagData;
    private int runFlag, nextFlag;
    //当 flagData 中的 data 值为 runFlag 时运行, 之后将其改为 nextFlag
    private char[] outputData;
    public void start(){ t.start(); }
    public T(MySignal d, char[] outPt, int x, int y){
        flagData=d; outputData=outPt; t=new Thread(this);
        runFlag=x; nextFlag=y;
    }
    public void run(){
        /* 请补充完整, 使之能得到期望的结果 */
    }
}
class App{
    public static void main (String[] args) {
        MySignal data=new MySignal(); T t1,t2,t3;
        char []c1={'1','2','3','4','5'};
        char []c2={'A','B','C','D','E'};
        char []c3={'甲','乙','丙','丁','戊'};
        t1=new T(data,c1,0,1); t2=new T(data,c2,1,2); t3=new T(data,c3,2,0);
        t1.start(); t2.start(); t3.start();
    }
}
```

输出结果为:

```
1 A 甲 2 B 乙 3 C 丙 4 D 丁 5 E 戊
```

二、简答题 (每小题 10 分, 共计 30 分)

- 1、简单说明 java 的异常处理策略。
- 2、简述结构化程序设计方法和面向对象程序设计方法的核心思想。
- 3、什么是设计的可维护性, 简要说明其对软件设计的重要意义。

三、综合设计题 (每小题 10 分, 共计 40 分)

注: 本大题答题时, 需要在答题纸上撰写全部代码

1、给定单链表类 `LinkedList`，请根据要求补充完成相关序列化和反序列化代码。其中 `App` 类中的有 (1) 处空需要填写。

```
import java.io.*;
import java.util.Scanner;
class LinkedList (1) {
    public LinkedList(){...//借助输入创建链表，无需补充；}
    public void showInfo(){...//从头至尾打印链表各结点值，无需补充;}
}
class App{
    public static void main (String[] args) throws Exception{
        LinkedList L,newL;
        L=new LinkedList();//假设输入 1 2 3 4 -1
        System.out.print("序列化前，链表为："); L.showInfo();
        System.out.print("\n 序列化……");
        /* 请将链表 L 借助序列化机制写入文件 L.dat */

        /* 请借助反序列化机制将文件 L.dat 中的链表复原，表头名称为 newL */
        System.out.print("反序列化后，链表为："); new_L.showInfo();
    }
}
```

输出结果为：

输入一组数创建链表，-1 表示结束：1 2 3 4 -1
序列化前，链表为：1 2 3 4
序列化……反序列化后，链表为：1 2 3 4

2、某云计算平拟向客户提供计算服务策略如下：向用户提供 `MyApp` 类，该类提供计算服务：`void compute()`。`MyApp` 类中还包含私有的用户信息 `name`，以及验证方法 `verify()`。客户向云计算平台提交包含自己的用户信息和计算方法。下面程序模拟上述过程，其中客户 A 的计算方法输出信息“我来试试”；客户 B 的计算方法输出信息“到此一游”。请基于**抽象类**，构造一组类，使之能得到给定的输出结果。其中 `App` 类中的有 (1)、(2) 两处空需要填写。

```
class UserInfo{;}//用户信息
abstract class MyApp{
    private UserInfo user;
    (1)boolean verify(){/*验证 user 是否为合法用户*/
        return true;
    } //请为 verify 方法添加合适的修饰，使得用户无法更改
}
class App{
    public static void runCompute((2) m){ m.compute();}
```

```
public static void main (String[] args) {
    UserInfo u1,u2;      Client_A a; Client_B b;
    u1=new UserInfo();  a=new Client_A(u1);runCompute(a);
    u2=new UserInfo();  b=new Client_B(u2);runCompute(b);
}
}
```

输出结果为：

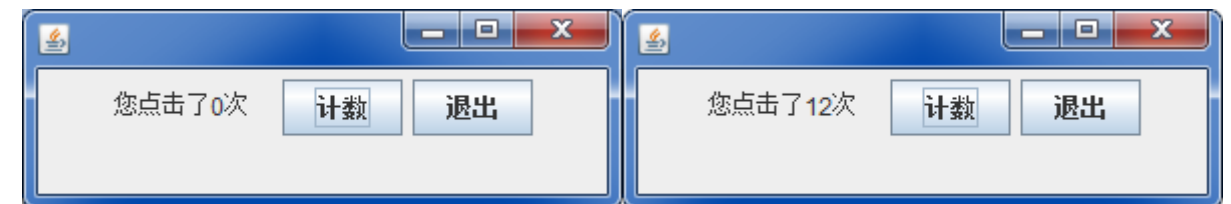
到此一游！
我来试试！

3、创建一个三角形类，包含属性：`private int a,b,c`；分别代表三角形的三条边。设计是需要满足如下需求： a. 在创建对象输入三条边，三边取值必须合法（三边均为正值，且任意两边之和大于第三边），否则将无法创建对象； b. 为使类更易于维护，必须单独设计一个 `boolean limit(...)` 函数，实现对三条边的限制检查，符合创建条件则返回真，否则返回假。 c. 该类有一个 `public void setEdges(int x, int y, int z)` 方法，将 `a/b/c` 的值替换成 `x/y/z`。但当 `x,y,z` 的值不满足三角形限制条件时，将不予替换，直接返回； d. 设计该三角形类，其中包含构造三角形类对象的手段。

```
class SanJiao{
}
```

4、程序运行初始界面如下左图，界面如下图所示。每次点击“计数”按钮，均会在相关标签中给出计数提示；点击退出按钮，则会结束程序。请补充完成此程序。另外，需要指明必须要导入的包，并填写 (1) 处的空

提示：`String.valueOf(10)` 可以将 10 转换成 `String` 型“10”



a.) 程序运行的初始界面

b.) 点击“计数”按钮 12 次后的界面

```
/* 请补充需要导入的包 */
class MyGUI extends JFrame (1){
    private JButton b_count,b_exit;
    private Label cLa; private int count;
    public MyGUI() {
        ..... //构造界面部分，无需填写
        //以上为 GUI 界面设计部分
        count=0;
    }
    public static void main(String[] args) { new MyGUI(); }
```