

5-1 计算整数序列和 分数 20

作者 huangcheng 单位 西南石油大学

以下程序计算多组整数序列和并输出：首先输入序列个数及每个序列中的整数个数，然后输入每个序列中的整数，最后计算每个序列的和值并逐行输出。

```
import java.util.*;
//线程类：计算整数序列和

class SumList implements Runnable { 4 分 {
    private ArrayList lt=null;//存放整数序列的列表
    SumList(ArrayList lt){
        this.lt=lt;
    }
    //线程体
    public void run() {
        int sum=0;
        for (int i = 0; i < lt.size(); i++)
            sum += (int) lt.get(i); 4 分 //取出序列元素累加
        System.out.println(sum);
    }
}
//测试类
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n=sc.nextInt();//输入序列个数
        int m=sc.nextInt();//输入每个序列中的整数个数
        for (int i = 0; i < n; i++) {
            ArrayList score = new ArrayList(); 4 分 //创建求和列表对象
            for (int j = 0; j < m; j++) {
                score.add(sc.nextInt());
            }
            SumList sl = new SumList(score); 4 分
            Thread th=new Thread(sl);//创建线程对象
            th.start(); 4 分 //启动线程
        }
        sc.close();
    }
}
```

部分正确: 16 分

5-2 利用线程同步机制执行正确的计数 分数 6

作者 huangcheng 单位 西南石油大学

当用多个线程处理共享变量时，线程中对共享变量的处理代码应用同步机制进行保护，才能保证处理的正确性。

```
import java.util.*;
class BackCounter implements Runnable{
    private int count=100; //线程共享变量，对它的处理必须用同步机制进行保护
    public int getCount() { return count; } //返回变量值
    //线程体
    public void run() {
        for(int i=10;i>0;i--) { //变量值递减 10
            synchronized (this) 2 分 { //以下代码在处理共享变量，需要同步机制保护
                if( count<=0 ) break;
                count--;
            }
            try { Thread.sleep(10); } catch ( InterruptedException e ) { } //模拟延时 10 毫秒
        }
    } //线程体结束
}
public class Main {
    public static void main(String[] args) throws InterruptedException { //某些线程方法会抛出检查型异常
        ArrayList<Thread> lt=new ArrayList<Thread>();
        BackCounter bc=new BackCounter();//创建实现类对象
        lt.add(new Thread(bc)); //创建线程对象
        lt.add(new Thread(bc));
        for (Thread th:lt) th.start() 2 分 ; //启动线程
        for (Thread th:lt) th.join() 2 分 ; //等待线程结束
        System.out.println(bc.getCount());
    }
}
```

答案正确: 6 分