

Java 组考核

姓名：

Tips:

- 1、 先将姓名填写在上面哦；
- 2、 本次考核内容源于培训所讲内容，但也会有培训之外的额外内容；
- 3、 本次考试时间共两个半小时；
- 4、 做完笔试部分前不允许开启计算机，提交笔试部分后方可开启计算机做机试部分。

笔试部分（50 分）

一、简答题

- 1 如何理解抽象类（4 分）
- 2 如何理解接口（4 分）
- 3 接口与抽象类之间有什么联系，有什么区别？（6 分）
- 4 ArrayList, HashMap, TreeSet 这三个集合类分别有什么特点，他们又有什么区别？（8 分）

- 5 阅读以下程序，其将会输出什么？（4 分）

```
class A {
    static void foo(){
        System.out.println("A.foo()");
    }
}

class B extends A{
    static void foo() {
        System.out.println("B.foo()");
    }
}

public class Main {
    public static void main(String[] args) {
        A a = new A();
        B b = new B();
        a.foo();
        b.foo();
    }
}
```

- 6 阅读以下程序，其将会输出什么？（4 分）

```
class A {
    void foo(){
        System.out.println("A.foo()");
    }
}

class B extends A{
    void foo() {
        System.out.println("B.foo()");
    }
}

public class Main {
    public static void main(String[] args) {
        A a = new A();
    }
}
```

```
        B b = new B();
        a.foo();
        b.foo();
    }
}
```

7 阅读以下代码，其运行后会输出什么？（4分）

```
import java.util.Objects;

class Student {
    private int id;
    private String name;

    public Student(int id, String name) {
        this.id = id;
        this.name = name;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Student student = (Student) o;
        return id == student.id &&
            Objects.equals(name, student.name);
    }

    @Override
    public int hashCode() {

        return Objects.hash(id, name);
    }

    @Override
    public String toString() {
        return "Student{" +
            "id=" + id +
            ", name='" + name + '\'' +
            '}';
    }
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Student student = new Student(20152727, "金子琦");  
        Student student2 = new Student(20152727, "金子琦");  
        System.out.println(student == student2);  
        System.out.println(student.equals(student2));  
    }  
}
```

8 如何理解类的继承 (6 分)

9 解释覆写和重载 (6 分)

10 调用下程序的 Test.fun()会输出 (发生) 什么。(4 分)

```
1  package one;  
2  
3  public class Test {  
4      public static void fun() {  
5          new LCM().f(4, 6);  
6      }  
7  }  
8  
9  class LCM extends GCD {  
10     @Override  
11     public int f(int a, int b) {  
12         int c = super.f(a, b);  
13         return a*b/c;  
14     }  
15 }  
16  
17 class GCD {  
18     public int f(int a, int b) {  
19         return b == 0 ? a : f(b, b:a % b);  
20     }  
21 }
```

机试部分（50 分）

姓名：

Tips:

- 1、先将姓名填写在上面哦；
- 2、提交笔试部分后方可作答此部分；
- 3、本部分评分根据要求实现程度、代码结构以及实现细节决定。

二、编程题

1 算法题（20 分）

时间限制：1s（C/C++），2s（Java） 内存限制：64MB

钊姐有一个数组，他上班无聊的时候就会往里面放一些数字。

钊姐有一天突发奇想，他想知道他都往这个数组里面放过那些数字，还想知道这些数字的和是多少。

不过钊姐上班（写 bug）很忙（很多），所以就拜托你来帮他写一个程序来计算这些。

为了简化操作，我们将其定义为三种操作：

- 1 输入 1 m：添加一个数字 m；
- 2 输入 2 1：按升序输出添加过的数字（不重复），若数组为空则输出 -1；
- 3 输入 2 2：输出添加过的数字的和。

输入描述：

第一行为一个整数 n，代表操作的总数。

后面 n 行，每行两个整数 k m，代表上述操作。

输出描述：

对于每一行输入 k 为 2 的情况，输出一行相应的输出（行末空格将被忽略）

备注：

$0 < n \leq 200000$

$0 \leq m < 10000$

操作 2 的出现次数不会超过 10 次

操作 3 的出现次数不会超过 10000 次

示例 1:

输入

```
5
1 2
1 3
1 2
2 1
2 2
```

输出

```
2 3
7
```

2 Java 编程题 (30 分)

要求:

1. 实现一个杂货铺 (Shop), 能够售卖多种商品 (Commodity);
2. 商品分为多种, 例如食品 (Food), 饮料 (Drink), 文具 (Stationery) 等。商品可以被售卖;
3. 杂货铺能够统计当前的营业额,
4. 要求合理抽象, 类结构符合逻辑。