

西 安 电 子 科 技 大 学

考试时间 120 分钟

试 题

题号	一	二	三	四	总分
分数					

1. 考试形式：闭卷 ☒ 开卷 ☐ ； 2. 本试卷共四大题，满分 100 分；
3. 考试日期： 2022 年 月 日；（答题内容请写在装订线外）

一、简答题（每题 4 分，共 16 分）

1. 简述定义类时实现接口和继承抽象类的代码结构及其主要异同。
2. 简述 protected 关键字是否可以修饰什么，以及可修饰时的作用。
3. 简述管道流概念和作用。

装订线装订线装订线

班级：姓名：学号：任课教师：

4. 解释说明以下 Java 代码片段分别采用（1）与（2）的功能以及区别。

```
Integer[] nums = {5,5,9,9,6,6,8,8,3,3,3,3,7,7,7};  
// (1)  
HashSet<Integer> set = new HashSet<>(Arrays.asList(nums));  
System.out.println(set);  
// (2)  
TreeSet<Integer> set2 = new TreeSet<>(Arrays.asList(nums));  
System.out.println(set2);
```

二、读程题（每题 10 分，共 40 分）

1.

```
class AnotherString {  
    public String anotherString(String s) {  
        if (s == null || s.length() == 0) {  
            return "";  
        }  
        String[] arr = s.split(" ");  
        StringBuilder sb = new StringBuilder();  
        for (int i = arr.length - 1; i >= 0; --i) {  
            if (!arr[i].equals("")) {  
                sb.append(arr[i]).append("\n");  
            }  
        }  
        return sb.length() == 0 ? "" : sb.substring(0, sb.length() - 1);  
    }  
    public static void main(String[] args) {  
        String str = "Learning Java is happy thing";  
        AnotherString rw = new AnotherString();  
        String anotherStr = rw.anotherString(str);  
        System.out.print(anotherStr);  
    }  
}
```

程序输出结果:

2.

```
class Base {
    void f() { System.out.println("1"); }
    Base() { f(); System.out.println("Base()"); }
}
class Child extends Base {
    private int r = 2;
    Child(int r) {
        this.r = r;
        System.out.println("(2) r = " + r);
    }
    @Override
    void f() {
        System.out.println("(3) r = " + r);
    }
}
public class Test {
    public static void main(String[] args) {
        Base c = new Child(22);
        c.f();
    }
}
```

程序输出结果:

3.

```
public class Test {  
    interface IntegerOperation {  
        int operation(int a, int b);  
    }  
    public int operate (int a, int b, IntegerOperation op) {  
        return op.operation(a, b);  
    }  
    public static void main(String... args) {  
        Test c = new Test();  
        IntegerOperation op1 = (a, b) -> a + b;  
        IntegerOperation op2 = (a, b) -> a - b;  
        System.out.println( c.operate (20, 2, op1) );  
        System.out.println( c.operate (20, 2, op2));  
  
        System.out.println( c.operate (20, 2, (a, b) -> a * b));  
    }  
}
```

程序输出结果:

4.说明编译运行以下 Java 代码的输出结果，以及交换（1）和（2）所示两行代码后的输出结果，并分别解释。

```
class Test {  
    int i = f();    // (1)  
    int j = g(i);  // (2)  
    int f() { return 22; }  
    int g(int n) { return n * 5; }  
    public static void main(String[] args)  
    {  
        Test test = new Test();  
        System.out.println(test.j);  
    }  
}
```

程序输出： _____

解释：

交换（1）和（2）所示两行代码后输出： _____

解释：

三、改错题（共 10 分）

下面 Java 程序中存在五处编译错误，指出错误位置，说明错误原因并更正。

1	import java.util.*;	解答：
2	public class SetTest {	
3	public static void main(String[] args) {	
4	Set<char> set = new Set<>();	
5	set.add('J');	
6	set.add('a');	
7	set.add('v');	
8	set.add('a');	
9		
10	Character[] chars = new Character[set.length];	
11	// 把集合转成数组	
12	set.toArray(chars);	
13	char max;	
14	for(char[] c : chars){	
15	if (c > max)	
16	max = c;	
17	}	
18	System.out.println("alphabet max: " + max);	
19	}	
20	}	

四、编程题（共 34 分）

1.（16 分）编写程序，读入 Words.txt，其中包含若干英文单词，每个单词一行。

控制台输出不重复单词的个数，并将不重复的单词按照字母序排序，将排序结果写入 Result.txt，每个单词一行。

2. (18 分) 按题目要求完成编程。

1) 设计线程安全的账号类 Account。

1.1) 该类包含字符串类型的成员变量 username 即用户名, 整数的成员变量 balance, 即账户余额。

1.2) 该类定义构造器带有一个 String 类型参数用户名;

1.3) 该类包含线程安全的 deposit 和 withdraw 方法, 参数分别为存/取款的额度, 以及 getBalance() 方法获取余额;

1.4) 当余额不足取款额度时, withdraw 方法抛出 OutOfBalanceException (假设已存在);

2) 创建线程类 DepositTransation 和 WithdrawTransation 模拟从账户存取款。

2.1) 其包含一个 Account 类型成员变量 account, 整型变量 amount 表示存/取款额度;

2.2) 构造器接收账号和存/取款额度;

2.3) 线程中执行存/取款操作。

3) 构建测试类 TransactionTest。

3.1) 在 main 方法中创建 Account 类对象, 设置初始用户名和余额(如 100);

3.2) 创建 3 个 DepositTransation 线程和 3 个 WithdrawTransation 线程并启动, 每个存款线程存入 1000 元; 每个取款线程取出 1200 元。

