

Java方向编程题答案

day19

[编程题]25089 -子串判断

链接: <https://www.nowcoder.com/questionTerminal/917a800d4de1423394827932f4725c68>

【题目解析】

无

【解题思路】

该题目描述看似非常复杂，实际上问题难点并不是特别大。化繁为简，题目表达的意思就是一个字符串数组p中每一个元素是否是字符串s的子串，将字符串数组p中每个元素遍历判断的结果true或者false组成数组返回。

【示例代码】

```
import java.util.*;

public class Substr {
    public boolean[] chkSubStr(String[] p, int n, String s) {
        // 创建boolean数组
        boolean[] hasSub=new boolean[p.length];
        //遍历数组 使用集合中特有功能
        for (int i = 0; i < p.length; i++) {
            hasSub[i]=s.contains(p[i]);
        }
        return hasSub;
    }
}
```

[编程题]36892-成绩排序

链接: <https://www.nowcoder.com/questionTerminal/0383714a1bb749499050d2e0610418b1>

【题目解析】

该题目主要是考察数据的排序。

【解题思路】

- 定义类，表示姓名和成绩
- 实现方法中：先输入人数，再输入排序策略（0或者1），输入多组人名和成绩
- 按照排序策略进行排序输出

【示例代码】

```
import java.util.ArrayList;
import java.util.Comparator;
import java.util.List;
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        while (scanner.hasNext()) {
            //输入人数
            int numPeople = scanner.nextInt();
            //排序策略
            int option = scanner.nextInt();

            //输入人员的姓名和成绩
            List<Student> stuList = new ArrayList<>();
            for (int i = 0; i < numPeople; i++) {
                stuList.add(new Student(scanner.next(), scanner.nextInt()));
            }

            //降序
            if (option == 0) {
                stuList.sort((o1, o2) -> o2.score - o1.score);
            } else if (option == 1) //升序
            {
                stuList.sort((o1, o2) -> o1.score - o2.score);
            }
            for (Student student : stuList) {
                System.out.println(student.name + " " + student.score);
            }
        }
    }

    class Student {
        public String name;
        public int score;

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