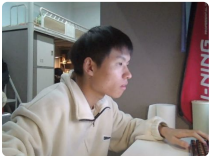


比特就业课Java方向笔试强训48天day20_11月8日（105Java班）-王世国-测评结果

考生信息  存在作弊行为



王世国
投递编号：91 | 学校：武汉轻工大学 | 邮箱：1477649017@qq.com | 职位：比特就业课105期Java2班
参考区域: 湖北武汉 (113.57.53.192) | 做题用时：01:01:35(2022-11-07 19:17:51开始答题，20:19:33交卷) |
作答设备：PC | 已同意诚信声明和隐私协议

100.0 分 / 100分

在本次考试中，考生总成绩为100.0分/100分，评级为**A（排名前1%）**，编程能力**优秀（2题完全通过，分数排名前1%）**，编程思路**完全正确**，编程规范性**高**。该考生在本次考试中**存在**作弊行为，和较早通过的代码相似度高（88.0%），视频监控图片数目过少，有遮挡或关闭摄像头的嫌疑。

考生成绩



题型	得分	正确题数	排名	用时	是否阅卷
单选	50.0	10	1	00:26:34	已阅
编程	50.0	2	1	00:33:58	已阅

作弊风险

高风险 未开启摄像头 考生可能关闭摄像头，导致视频截图较少	低风险 代码流出 编程题2 被1人抄袭	高风险 代码相似度 编程题2 88.00%
---	-------------------------------	--

知识点技能图谱

其他知识点	1/1
树	3/3
递归	1/1
排序	1/1
字符串	2/2
堆	1/1
栈	3/3
链表	1/1
队列	2/2
哈希	1/1

知识点	得分	正确题数
其他知识点	25.0	1
树	15.0	3
递归	5.0	1
排序	5.0	1
字符串	30.0	2
堆	5.0	1

知识点	得分	正确题数
栈	15.0	3
链表	5.0	1
队列	10.0	2
哈希	5.0	1

历史笔试记录

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	安排笔试时间	交卷时间
1	比特就业课105期+2022寒假班C1考试	20.0%	48.8/60	单选:30.0分 编程:18.75分	否	2022-03-29 11:16:18	2022-03-31 18:51:27
2	比特就业课105期+2022寒假班C2考试	66.0%	24.0/60	单选:24.0分 编程:0.0分	否	2022-04-11 14:12:23	2022-04-11 20:12:16
3	比特就业课 105期JavaSE考试	11.0%	56.0/60	单选:26.0分 编程:30.0分	否	2022-07-12 16:00:16	2022-07-13 15:48:42
4	比特就业课 105期java方向 数据结构考试	47.0%	50.0/60	单选:20.0分 编程:30.0分	否	2022-07-23 12:49:22	2022-07-25 09:56:28
5	比特就业课105期Java方向笔试强训48天 day01_10月10日	36.0%	80.0/100	单选:40.0分 编程:40.0分	是，摄像头监控异常	2022-10-09 17:29:16	2022-10-09 21:57:25
6	比特就业课105期Java方向笔试强训48天 day02_10月11日	13.0%	90.0/100	单选:40.0分 编程:50.0分	是，代码抄袭	2022-10-10 10:43:48	2022-10-10 21:13:15
7	比特就业课105期Java方向笔试强训48天 day03_10月12日	1.0%	95.0/100	单选:45.0分 编程:50.0分	是，摄像头监控异常	2022-10-11 10:40:53	2022-10-12 10:03:09
8	比特就业课105期Java方向笔试强训48天 day04_10月13日	2.0%	95.0/100	单选:30.0分 不定项选择:15.0分 编程:50.0分	是，摄像头监控异常	2022-10-12 10:31:10	2022-10-12 21:14:05
9	比特就业课105期Java方向笔试强训48天 day05_10月14日	12.0%	85.0/100	单选:35.0分 不定项选择:0.0分 编程:50.0分	是，摄像头监控异常 代码抄袭	2022-10-13 11:41:43	2022-10-14 11:33:24
10	比特就业课105期Java方向笔试强训48天 day06_10月15日	13.0%	86.7/100	单选:25.0分 不定项选择:11.67分 编程:50.0分	是，摄像头监控异常	2022-10-14 10:59:38	2022-10-14 21:18:11
11	比特就业课105期Java方向笔试强训48天 day07_10月17日	1.0%	100.0/100	单选:50.0分 编程:50.0分	是，摄像头监控异常	2022-10-16 16:46:53	2022-10-16 20:38:39
12	比特就业课105期Java方向笔试强训48天 day08_10月18日	33.0%	87.5/100	单选:40.0分 编程:47.5分	是，摄像头监控异常	2022-10-17 16:18:42	2022-10-17 20:16:45

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	安排笔试时间	交卷时间
13	比特就业课105期Java方向笔试强训48天 day09_10月19日	3.0%	95.0/100	单选:45.0分 编程:50.0分	是，摄像头监控异常	2022-10-18 17:07:17	2022-10-19 11:02:55
14	比特就业课105期Java方向笔试强训48天 day10_10月20日	58.0%	65.0/100	单选:40.0分 编程:25.0分	是，摄像头监控异常	2022-10-19 15:29:54	2022-10-19 21:44:34
15	比特就业课105期Java方向笔试强训48天 day11_10月21日	2.0%	95.0/100	单选:45.0分 编程:50.0分	是，摄像头监控异常	2022-10-20 16:46:09	2022-10-20 23:36:15
16	比特就业课105期Java方向笔试强训48天 day12_10月22日	15.000001%	90.0/100	单选:40.0分 编程:50.0分	是，摄像头监控异常	2022-10-21 16:44:18	2022-10-21 22:49:24
17	比特就业课105期Java方向笔试强训48天 day13_10月24日	31.0%	65.0/100	单选:40.0分 编程:25.0分	是，摄像头监控异常 代码抄袭	2022-10-22 16:07:25	2022-10-23 23:15:22
18	比特就业课105期Java方向笔试强训48天 day14_10月25日	58.999996%	55.0/100	单选:30.0分 编程:25.0分	是，摄像头监控异常	2022-10-24 14:08:32	2022-10-24 23:42:01
19	比特就业课105期Java方向笔试强训48天 day15_10月26日	42.0%	55.0/100	单选:30.0分 编程:25.0分	是，摄像头监控异常	2022-10-24 14:11:25	2022-10-25 22:58:05
20	比特就业课105期Java方向笔试强训48天 day16_10月27日	13.0%	89.4/100	单选:45.0分 编程:44.44分	是，摄像头监控异常	2022-10-24 14:25:44	2022-10-26 21:33:12
21	比特就业课105期Java方向笔试强训48天 day17_10月28日	24.0%	83.3/100	单选:40.0分 编程:43.33分	是，摄像头监控异常	2022-10-24 14:29:13	2022-10-28 22:18:13
22	比特就业课105期Java方向笔试强训48天 day18_10月29日	12.0%	84.1/100	单选:40.0分 编程:44.12分	是，摄像头监控异常	2022-10-24 14:31:43	2022-10-29 15:39:41
23	比特就业课Java方向Java班笔试强训48天 day19_11月7日（105Java班）	47.0%	72.5/95	单选:40.0分 编程:32.5分	是，摄像头监控异常	2022-11-06 14:19:57	2022-11-07 11:29:43

编码能力

题号	正确性	提交次数	做题用时	使用语言	运行时间	占用内存	编程思路	代码规范	成绩排名
编程题1	100%	3	00:06:37	Java	47ms	10808K	良	良	1%
编程题2	100%	1	00:27:21	Java	44ms	11248K	良	良	1%

1

[单选题 | 平均分3.02分 | 52人正确/86人做题 | 用时：<1分]

得分：5.0 / 5.0

对于顺序存储的线性表，访问结点和增加结点的时间复杂度为（ ）。

- A $O(n)$ $O(n)$
- B $O(n)$ $O(1)$
- C $O(1)$ $O(n)$
- D $O(1)$ $O(1)$

他的回答： C (正确)

正确答案： C

2 [单选题 | 平均分2.47分 | 43人正确/87人做题 | 用时：4分 | 得分：5.0 / 5.0

若栈采用顺序存储方式存储，现两栈共享空间 $V[1..m]$ ， $top[i]$ 代表第 i 个栈($i=1,2$)栈顶，栈1的底在 $v[1]$ ，栈2的底在 $V[m]$ ，则栈满的条件是()。

- A $top[1]+top[2]=m$
- B $top[1]+1=top[2]$
- C $top[2]-top[1]=0$
- D $top[1]=top[2]$

他的回答： B (正确)

正确答案： B

3 [单选题 | 平均分4.47分 | 76人正确/85人做题 | 用时：<1分 | 得分：5.0 / 5.0

下述有关栈和队列的区别，说法错误的是？

- A 栈是限定只能在表的一端进行插入和删除操作。
- B 队列是限定只能在表的一端进行插入和在另一端进行删除操作。
- C 栈和队列都属于线性表
- D 栈的插入操作时间复杂度都是 $o(1)$ ，队列的插入操作时间复杂度是 $o(n)$

他的回答： D (正确)

正确答案： D

4 [单选题 | 平均分4.48分 | 77人正确/86人做题 | 用时：<1分 | 得分：5.0 / 5.0

从前有座山，山里有座庙，庙里有个老和尚，再给小和尚讲故事，故事内容是：从前有座山，山里有座庙，庙里有个老和尚，再给小和尚讲故事，故事内容是：从前有座山，山里有座庙，庙里有个老和尚，再给小和尚讲故事，故事内容是.....描述的是()

- A 贪心
- B 回溯
- C 穷举
- D 分治
- E 递归

他的回答： E (正确)

正确答案： E

参考答案：

递归指的是一个过程：函数不断引用自身，直到引用的对象已知。

5 [单选题 | 平均分3.33分 | 58人正确/87人做题 | 用时：4分 | 得分：5.0 / 5.0

某二叉树共有 399 个结点，其中有 199 个度为 2 的结点，则该二叉树中的叶子结点数为()

- A 不存在这样的二叉树
- B 200
- C 198
- D 199

他的回答： B (正确)

正确答案： B

参考答案：

根据二叉树的基本性质，对任何一棵二叉树，度为 0 的结点（即叶子结点）总是比度为 2 的结点多一个。题目中度为 2 的结点为 199 个，则叶子结点为 $199+1=200$ 。故本题答案为 B 选项。

6 [单选题 | 平均分4.24分 | 72人正确/85人做题 | 用时：<1分 | 得分：5.0 / 5.0

某二叉树的前序遍历序列与中序遍历序列相同，均为 ABCDEF，则按层次输出（同一层从左到右）的序列为（ ）

- A ABCDEF
- B BCDEFA
- C FEDCBA
- D DEFABC

他的回答：A (正确)

正确答案：A

参考答案：

二叉树遍历可以分为 3 种：前序遍历（访问根结点在访问左子树和访问右子树之前）、中序遍历（访问根结点在访问左子树和访问右子树两者之间）、后序遍历（访问根结点在访问左子树和访问右子树之后）。

二叉树的中序遍历序列和前序遍历序列均为 ABCDEF，可知该树只有右子树结点，没有左子树结点，A 为根结点，因此该树有 6 层，从顶向下从左向右依次为 ABCDEF。故本题答案为 A 选项。

7 [单选题 | 平均分3.65分 | 62人正确/85人做题 | 用时：2分 | 得分：5.0 / 5.0

初始序列为1 8 6 2 5 4 7 3—组数采用堆排序，当建堆（小根堆）完毕时，堆所对应的二叉树中序遍历序列为（ ）

- A 8 3 2 5 1 6 4 7
- B 3 2 8 5 1 4 6 7
- C 3 8 2 5 1 6 7 4
- D 8 2 3 5 1 4 7 6

他的回答：A (正确)

正确答案：A

参考答案：

答案：选A

答案解析：初始化序列：1 8 6 2 5 4 7 3，小根堆就是要求结点的值小于其左右孩子结点的值，左右孩子的大小没有关系，那么小根堆排序之后为：1 2 4 3 5 6 7 8；

中序遍历：先遍历左孩子，然后访问根结点，最后访问有孩子，故遍历结果为：8 3 2 5 1 6 4 7

8 [单选题 | 平均分3.53分 | 60人正确/85人做题 | 用时：<1分 | 得分：5.0 / 5.0

解决散列法中出现冲突问题常采用的方法是（ ）

- A 数字分析法、除余法、平方取中法
- B 数字分析法、除余法、线性探测法
- C 数字分析法、线性探测法、多重散列法
- D 线性探测法、多重散列法、链地址法

他的回答：D (正确)

正确答案：D

9 [单选题 | 平均分2.12分 | 36人正确/85人做题 | 用时：10分 | 得分：5.0 / 5.0

以下哪种排序算法对(1,3,2,4,5,6,7,8,9)进行的排序最快（ ）

- A 冒泡
- B 快排
- C 归并
- D 堆排

在这种区域有序的情况下，使用优化版的冒泡是比较好的，比较几次就可以对于归并和堆排时间复杂度是 $n \times \log n$ ，快速排序会趋于 n^2

他的回答：A (正确)

正确答案：A

10 [单选题 | 平均分4.35分 | 74人正确/85人做题 | 用时：<1分] 得分：5.0 / 5.0

以下数据结构中，（ ）是非线性数据结构

- A 树
- B 字符串
- C 队
- D 栈

他的回答：A (正确)

正确答案：A

11 ACM编程题 语言限制 [编程题 | 平均分24.11分 | 81人正确/84人做题 | 提交: 3 次] 得分：25.0 / 25.0

标题：字符串反转 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【字符串反转】

接受一个只包含小写字母的字符串，然后输出该字符串反转后的字符串。（字符串长度不超过1000）

输入描述：

输入一行，为一个只包含小写字母的字符串。

输出描述：

输出该字符串反转后的字符串。

示例1：

输入

abcd

输出

dcba

代码片段

功能实现			代码提交统计			代码执行统计	
总通过率	TA的	平均		TA的	平均	答案错误：2 答案正确：1	
	100%	96%	使用语言	Java			
基本测试用例通过率	7/7	96%	做题用时	00:06:37	00:06:56		
	(100%)		提交次数	3	1		
边缘测试用例通过率	4/4	96%					

他的代码：

做题用时: 6 分钟 语言：Java 运行时间：47ms 占用内存：10808K 程序状态：答案正确

```
import java.util.*;
public class Main {
    public static void main(String[] args){
```

```

Scanner scan = new Scanner(System.in);
StringBuilder stringBuilder = new StringBuilder(scan.next());
int left = 0;
int right = stringBuilder.length() - 1;
while(left < right){
    char ch = stringBuilder.charAt(left);
    stringBuilder.setCharAt(left,stringBuilder.charAt(right));
    stringBuilder.setCharAt(right,ch);
    left++;
    right--;
}
System.out.println(stringBuilder.toString());
}
}

```



[点此](#)或手机扫描二维码查看代码编写过程

12 ACM编程题 语言限制 [编程题 | 平均分15.63分 | 49人正确/86人做题 | 提交: 1 次] 得分 : 25.0 / 25.0

标题：公共子串计算 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【公共子串计算】

给定两个只包含小写字母的字符串，计算两个字符串的最大公共子串的长度。

注：子串的定义指一个字符串删掉其部分前缀和后缀（也可以不删）后形成的字符串。

$$1 \leq s \leq 150$$

数据范围：字符串长度：

$$O(n^3)$$

$$O(n)$$

进阶：时间复杂度：，空间复杂度：

输入描述：

输入两个只包含小写字母的字符串

输出描述：

输出一个整数，代表最大公共子串的长度

示例1：

输入

asdfas werasdfaswer

输出

6

代码片段

功能实现

代码提交统计

代码执行统计

总通过率	TA的 100%	平均 62%	使用语言	TA的 Java	平均	答案正确 : 1
基本测试用例通过率	8/8 (100%)	63%	做题用时	00:27:21	00:17:22	
边缘测试用例通过率	5/5 (100%)	60%	提交次数	1	2	
代码效率			代码规范及可读性			
运行时间	TA的 44ms	参考 1s	代码规范得分			4.8
占用内存	11248K	32768K	Line 2: 'CLASS_DEF' should be separated from previous statement. [EmptyLineSeparator]			

他的代码：

做题用时: 27 分钟 语言 : Java 运行时间 : 44ms 占用内存 : 11248K 程序状态 : 答案正确

```
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        String str1 = scan.next();
        String str2 = scan.next();
        //保证str1中存放的是长度较小的字符串
        if(str1.length() > str2.length()){
            String tmp = str1;
            str1 = str2;
            str2 = tmp;
        }
        if(str2.contains(str1)){
            System.out.println(str1.length());
            return ;
        }
        int max = 0;
        for(int i = 0;i < str1.length();i++){
            for(int j = i + 1;j < str1.length();j++){
                String tmp = str1.substring(i,j+1);
                if(str2.contains(tmp)){
                    if(tmp.length() > max){
                        max = tmp.length();
                    }
                }
            }
        }
        System.out.println(max);
    }
}
```




[点此](#)或手机扫描二维码查看代码编写过程

监控截图



代码相似度

编程题2相似代码			
相似代码来源考生：黄林	试卷：比特就业课Java方向笔试强训48天day20_11月8日（105Java班）	相似度：88.00%	通过时间提前：00:34:30

本代码

```
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        String str1 = scan.next();
        String str2 = scan.next();
        //保证str1中存放的是长度较小的字符串
        if(str1.length() > str2.length()){
            String tmp = str1;
            str1 = str2;
            str2 = tmp;
        }
        if(str2.contains(str1)){
            System.out.println(str1.length());
            return ;
        }
        int max = 0;
        for(int i = 0;i < str1.length();i++){
            for(int j = i + 1;j < str1.length();j++){
                String tmp = str1.substring(i,j+1);
                if(str2.contains(tmp)){
                    if(tmp.length() > max){
                        max = tmp.length();
                    }
                }
            }
        }
        System.out.println(max);
    }
}
```

相似代码

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        String s1=sc.nextLine();
        String s2=sc.nextLine();
        int max=0;
        int n=0;
        if (s1.length()>s2.length()){
            String s=s1;
            s1=s2;
            s2=s;
        }
        for (int i=0;i<s1.length();i++){
            for (int j=i+1;j<=s1.length();j++){
                String ss=s1.substring(i,j);
                if (s2.contains(ss)){
                    if (ss.length()>max){
                        max=ss.length();
                    }
                }
            }
        }
        System.out.println(max);
    }
}
```

疑似代码流出

编程题2相似代码

相似代码来源考生：文国舟

试卷：比特就业课Java方向笔试强训48天day20_11月8日（105Java班）

相似度：80.41%

通过时间落后：27:40:17

本代码

```
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        String str1 = scan.next();
        String str2 = scan.next();
        //保证str1中存放的是长度较小的字符串
        if(str1.length() > str2.length()){
            String tmp = str1;
            str1 = str2;
            str2 = tmp;
        }
        if(str2.contains(str1)){
            System.out.println(str1.length());
            return ;
        }
        int max = 0;
        for(int i = 0;i < str1.length();i++){
            for(int j = i + 1;j < str1.length();j++){
                String tmp = str1.substring(i,j+1);
                if(str2.contains(tmp)){
                    if(tmp.length() > max){
                        max = tmp.length();
                    }
                }
            }
        }
        System.out.println(max);
    }
}
```

相似代码

```
import java.util.Scanner;

// 注意类名必须为 Main, 不要有任何 package xxx 信息
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String s = sc.next();
        String s1 = sc.next();
        if (s.length() >= s1.length()) {
            String temp = s;
            s = s1;
            s1 = temp;
        }
        int len = s.length();
        for (int i = s.length(); i > 0; i--) { //字符串长度
            for (int j = 0; j <= s.length() - i; j++) {
                String temp = s.substring(j, i + j);
                if (s1.contains(temp)) {
                    //System.out.println(temp);
                    System.out.println(temp.length());
                    return;
                }
            }
        }
        System.out.println(0);
    }
}
```