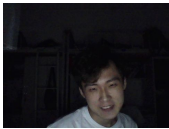


# 每日一题day20\_6月11日测评结果

## 考生信息



张博翔

考号：1675 | 学校：陕西科技大学 | 邮箱：1761607418@qq.com | 职位：43班 |

参考区域: 陕西省西安市 ( 111.114.0.2 ) | 做题用时：00:46:18(2019-06-11 00:22:00 - 01:08:28)

## 考生成绩



题型	得分	正确题数	排名	用时	是否阅卷
单选	45.0	9	7	00:22:43	--
编程	50.0	2	1	00:21:20	--

## 知识点技能图谱



知识点	得分	正确题数
腾讯	5.0	1
字符串	50.0	2
百度	5.0	1
堆	5.0	1
栈	10.0	2
队列	5.0	1
哈希	0.0	0
树	10.0	2
递归	5.0	1
排序	5.0	1
查找	25.0	1
链表	5.0	1
图	5.0	1

## 历史笔记录

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	笔试时间
1	每日一题day1_5月20日	4.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-05-19 16:08:49

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	笔试时间
2	每日一题day02_5月21日	51.0%	60.0/100	单选:30.0分 编程:30.0分	否	2019-05-20 17:40:56
3	每日一题day03_5月22日	4.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-05-21 12:59:15
4	每日一题day04_5月23日	8.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-05-22 10:40:18
5	每日一题day05_5月24日	29.0%	75.0/100	单选:25.0分 编程:50.0分	否	2019-05-22 20:39:16
6	每日一题day06_5月25日	10.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-05-24 10:41:21
7	每日一题day07_5月27日	6.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-05-26 15:09:41
8	每日一题day08_5月28日	18.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-05-27 14:06:31
9	每日一题day09_5月29日	13.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-05-28 15:11:53
10	每日一题day10_5月30日	6.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-05-29 11:32:20
11	每日一题day11_5月31日	14.0%	80.0/100	单选:30.0分 编程:50.0分	否	2019-05-30 11:22:09
12	每日一题day12_6月1日	31.0%	75.0/100	单选:25.0分 编程:50.0分	否	2019-05-31 10:33:31
13	每日一题day13_6月3日	9.0%	85.0/100	单选:35.0分 编程:50.0分	否	2019-06-02 15:20:12
14	每日一题day14_6月4日	41.0%	50.0/100	单选:25.0分 编程:25.0分	否	2019-06-03 11:53:13
15	每日一题day15_6月5日	6.0%	85.0/100	单选:35.0分 编程:50.0分	否	2019-06-04 12:02:48
16	每日一题day16_6月6日	6.0%	80.0/100	单选:30.0分 编程:50.0分	否	2019-06-05 16:32:52
17	每日一题day17_6月7日	4.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-06-06 10:57:01
18	每日一题day18_6月8日	16.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-06-07 15:47:43
19	每日一题day19_6月10日	3.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-06-09 11:55:46

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%	1	00:04:45	C++	3ms	476K	优	优	1%
编程题2	100%	1	00:16:35	C++	4ms	488K	优	优	1%

1 [平均分3.2分 | 68人正确/107人做题 | 用时 : 5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 [平均分3.1分 | 65人正确/104人做题 | 用时：2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  $\text{top}[1]+\text{top}[2]=m$
- B  $\text{top}[1]+1=\text{top}[2]$
- C  $\text{top}[2]-\text{top}[1]=0$
- D  $\text{top}[1]=\text{top}[2]$

他的回答： B (正确)

正确答案： B

3 [平均分4.5分 | 93人正确/104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 [平均分4.8分 | 101人正确/106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 [平均分3.0分 | 61人正确/102人做题 | 用时：2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 [平均分4.7分 | 95人正确/101人做题 | 用时：2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。中序遍历序列与前序遍历序列相同说明该树只有右子树没有左子树，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 [平均分3.8分 | 76人正确/99人做题 | 用时：4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：

解析：堆排序：利用堆的性质进行的一种选择排序 答案：A

8 [平均分2.3分 | 44人正确/97人做题 | 用时：<1分 | 得分：0.0 / 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： C (错误)

正确答案： D

9 [平均分2.1分 | 43人正确/102人做题 | 用时：2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

他的回答： A (正确)

正确答案： A

10 [平均分3.6分 | 71人正确/98人做题 | 用时：<1分 | 得分：5.0 / 5.0]

设无向图的顶点个数为n，则该图最多有多少条边？

- A n-1
- B  $n(n+1)/2$
- C  $n(n-1)/2$
- D n
- E 不同于以上答案

他的回答： C (正确)

正确答案：C

11 [平均分24.3分 | 102人正确/105人做题 | 提交: 1 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】

写出一个程序，接受一个字符串，然后输出该字符串反转后的字符串。例如：

输入描述：

输入N个字符

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	
	TA的	平均		TA的	平均	答案正确：1	
总通过率	100%	97%	使用语言	C++			
基本测试用例通过率	6/6 (100%)	97%	做题用时	00:04:45	00:10:24		
边缘测试用例通过率	4/4 (100%)	97%	提交次数	1	3		
代码效率					代码规范及可读性		
	TA的	参考				代码规范得分 5.0	
运行时间	3ms	1s					
占用内存	476K	32768K					

他的代码：

做题用时: 4 分钟    语言：C++    运行时间：3ms    占用内存：476K    程序状态：答案正确

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str1;
    while(cin >> str1)
    {
        string str2;
        for(int i = str1.size() - 1; i >= 0; i--)
        {
            str2 += str1[i];
        }
        cout << str2 << endl;
    }
}
```

标题：公共字符串计算 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【公共字符串计算】

题目标题：  
计算两个字符串的最大公共字符串的长度，字符不区分大小写  
详细描述：  
接口说明

原型：  
int getCommonStrLength(char \* pFirstStr, char \* pSecondStr);

输入参数：  
char \* pFirstStr //第一个字符串  
char \* pSecondStr//第二个字符串

输入描述：

输入两个字符串

输出描述：

输出一个整数

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 86%	使用语言	TA的 C++	平均	答案正确：1	
基本测试用例通过率	6/6 (100%)	86%	做题用时	00:16:35	00:31:14		
边缘测试用例通过率	4/4 (100%)	87%	提交次数	1	4		
代码效率					代码规范及可读性		
运行时间	TA的 4ms	参考 1s				代码规范得分 5.0	
占用内存	488K	32768K					

他的代码：

做题用时: 16 分钟    语言：C++    运行时间：4ms    占用内存：488K    程序状态：答案正确

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str1;
    string str2;
    while(cin >> str1 >> str2)
    {
        string curString;
        string maxString;
        for(int i = 0; i < str1.size(); i++)
        {
            if(str1[i] >= 'A' && str1[i] <= 'Z')
```

```

    {
        str1[i] -= ' ';
    }
}
for(int i = 0; i < str2.size(); i++)
{
    if(str2[i] >= 'A' && str2[i] <= 'Z')
    {
        str2[i] -= ' ';
    }
}
for(int i = 0; i < str1.size(); i++)
{
    for(int j = 0; j < str2.size(); j++)
    {
        if(str1[i] == str2[j])
        {
            int m = i;
            int n = j;
            while(str1[m] == str2[n] && str1[m] != '\0')
            {
                curString += str1[m];
                //cout << curString << endl;
                m++;
                n++;
            }
            if(curString.size() > maxString.size())
            {
                maxString = curString;
            }
            curString.resize(0);
        }
    }
}
//cout << maxString << endl;
cout << maxString.size() << endl;
}
}

```