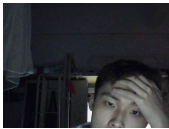


# 每日一题day28\_7月4日测评结果

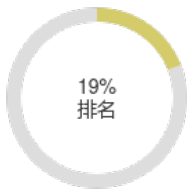
## 考生信息



张博翔

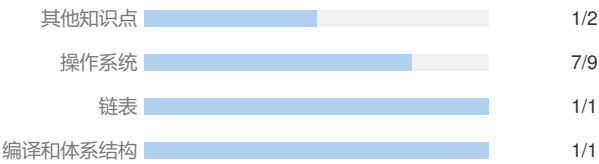
考号：1675 | 学校：陕西科技大学 | 邮箱：1761607418@qq.com | 职位：43班 |  
参考区域: 陕西省西安市 ( 111.114.0.2 ) | 做题用时：00:00:24(2019-07-04 00:27:53 - 02:10:05)

## 考生成绩



题型	得分	正确题数	排名	用时	是否阅卷
单选	60.0	8	14	00:00:21	--
编程	25.0	1	18	00:00:00	--

## 知识点技能图谱



知识点	得分	正确题数
其他知识点	25.0	1
操作系统	55.0	7
链表	25.0	1
编译和体系结构	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	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
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

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	笔试时间
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
20	每日一题day20_6月11日	4.0%	95.0/100	单选:45.0分 编程:50.0分	否	2019-06-10 14:43:47
21	每日一题day21_6月12日	5.0%	85.0/100	单选:35.0分 编程:50.0分	否	2019-06-11 11:37:10
22	每日一题day22_6月13日	6.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-06-12 11:18:48
23	每日一题day23_6月14日	1.0%	100.0/100	单选:50.0分 编程:50.0分	否	2019-06-13 11:26:22
24	每日一题day24_6月15日	7.0%	80.0/100	单选:30.0分 编程:50.0分	否	2019-06-14 12:03:39
25	每日一题day25_7月1日	6.0%	85.0/100	单选:35.0分 编程:50.0分	否	2019-06-30 11:46:11
26	每日一题day26_7月2日	9.0%	90.0/100	单选:40.0分 编程:50.0分	否	2019-07-01 14:57:52
27	每日一题day27_7月3日	23.0%	70.0/100	单选:20.0分 编程:50.0分	否	2019-07-02 11:54:51

编码能力



1

[平均分3.3分 | 63人正确/96人做题 | 用时 : <1分 | 得分 : 5.0 / 5.0]

下面的程序执行输出几个hello ?

```
#include<stdio.h>
#include <unistd.h>
int main( ) {
    fork( );
    fork( );
    fork( );
    printf("hello\n");
    return 0;
}
```

- B 4
- C 6
- D 8

他的回答： D (正确)

正确答案： D

2 [平均分3.4分 | 61人正确/91人做题 | 用时：<1分 | 得分：5.0 / 5.0

有一个程序中有A,B,C三个线程同时对一个文件进行读写操作，其中的A,B是写进程只负责往里面写数据，C是读线程，同时把读取的数据从文件中删除，A线程单独写满文件需要10个小时，B单独写程序需要6小时，C线程需要15小时才能读取完整个文件，不考虑三个线程之间的相互影响的情况下现在\_\_\_\_小时才能写满文件

- A 5
- B 6
- C 5.5
- D 4.5
- E 4.8
- F 5.3

他的回答： A (正确)

正确答案： A

3 [平均分4.0分 | 76人正确/96人做题 | 用时：<1分 | 得分：5.0 / 5.0

系统中内存不足程序所需大小，程序就无法执行。

- A 错
- B 对

他的回答： A (正确)

正确答案： A

4 [平均分3.7分 | 71人正确/96人做题 | 用时：<1分 | 得分：5.0 / 5.0

通常所说的"存储保护"的基本含义是（ ）。

- A 防止存储器硬件受损
- B 防止程序在内存丢失
- C 防止程序间相互越界访问
- D 防止程序被人偷看

他的回答： C (正确)

正确答案： C

5 [平均分3.5分 | 65人正确/94人做题 | 用时：<1分 | 得分：0.0 / 5.0

下列进程调度算法中，（ ）可能会出现进程长期得不到调度的情况。

- A 非强占式静态优先权法
- B 强占式静态优先权法
- C 时间片轮转调度算法
- D 非强占式动态优先权法

他的回答： D (错误)

正确答案： B

6 [平均分3.6分 | 68人正确/95人做题 | 用时：<1分 | 得分：5.0 / 5.0

如果信号量的当前值为 - 4，则表示系统中在该信号量上有（ ）个进程等待。

- A 4
- B 3
- C 5
- D 0

他的回答： A (正确)

正确答案： A

7 [平均分3.4分 | 63人正确/93人做题 | 用时：<1分 | 得分：5.0 / 5.0

设两个进程共用一个临界资源的互斥信号量mutex=1，当mutex = - 1时表示（ ）。

- A 一个进程进入了临界区，另一个进程等待
- B 没有一个进程进入临界区
- C 两个进程都进入临界区
- D 两个进程都在等待

他的回答： A (正确)

正确答案： A

8 [平均分2.0分 | 38人正确/94人做题 | 用时：<1分 | 得分：0.0 / 5.0

若系统中只有用户级线程，则处理机调度单位是()。

- A 线程
- B 进程
- C 程序
- D 作业

他的回答： A (错误)

正确答案： B

9 [平均分2.3分 | 44人正确/94人做题 | 用时：<1分 | 得分：5.0 / 5.0

一个在线服务器通常需要读取存储着海量数据的数据库。为了提高服务器处理速度，通常需要加cache（缓存），以下场景中不适合使用cache的是

- A 数据库中每条数据被访问的概率近似相等，且独立
- B 使用了多线程机制的服务
- C 单条线程尺寸太小的数据
- D 有着大量访问的服务

他的回答： A (正确)

正确答案： A

10 [平均分16.1分 | 61人正确/95人做题 | 用时：<1分 | 得分：25.0 / 25.0

若某线性表最常用的操作是存取任一指定序号的元素和在最后进行插入和删除运算，则利用（ ）存储方式最节省时间。

- A 顺序表
- B 双链表
- C 带头结点的双循环链表
- D 单循环链表

他的回答： A (正确)

正确答案： A

11 [平均分7.2分 | 24人正确/83人做题 | 提交：9次 | 得分：0.0 / 25.0

标题：奇数位上都是奇数或者偶数位上都是偶数 | 时间限制：1秒 | 内存限制：32768K | 语言限制：[C++, Java]

【奇数位上都是奇数或者偶数位上都是偶数】给定一个长度不小于2的数组arr。 写一个函数调整arr，使arr中要么所有的偶数位上都是偶数，要么所有的奇数位上都

是奇数上。要求：如果数组长度为N，时间复杂度请达到O(N)，额外空间复杂度请达到O(1),下标0,2,4,6...算作偶数位,下标1,3,5,7...算作奇数位，例如[1,2,3,4]调整为[2,1,4,3]即可

输入描述：

输出描述：

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		
总通过率		0%	28%	使用语言		C++		答案错误：3	
基本测试用例通过率		0/1 (0%)	28%	做题用时		00:00:00	00:40:28	运行超时：2	
				提交次数		9	7	编译错误：4	
代码效率							代码规范及可读性		
		TA的	参考					代码规范得分	
运行时间		3ms	1s					4.4	
占用内存		472K	32768K					Line 9: Is this a non-const reference? If so, make const or use a pointer: vector& arr [runtime/references] [2]	
								Line 21: Add #include for swap [build/include_what_you_use] [4]	
								Line 9: Add #include for vector<> [build/include_what_you_use] [4]	

他的代码：

做题用时: <1 分钟    语言：C++    运行时间：3ms    占用内存：472K    程序状态：答案错误

```
class Solution {
public:
    /**
     * 奇数位上都是奇数或者偶数位上都是偶数
     * 输入：数组arr，长度大于2
     * len：arr的长度
     * 将arr调整成奇数位上都是奇数或者偶数位上都是偶数
     */
    void oddInOddEvenInEven(vector<int>& arr, int len)
    {
        //思路：i指向偶数位，j指向奇数位，由于不清楚其中有多少偶数和奇数，因此我们要两边同时进行判断
        //只要奇数位全为奇数，偶数位全为偶数就都可以终止循环
        //如果发生当前所指奇数位为偶数，并且当前所指偶数位为偶数则进行交换
        //如果当前奇数位为奇数则指向朝后移动
        //如果当前偶数位为偶数则指向朝后移动

        int i = 0, j = 1;
        while(i < len && j < len)
        {
            if(arr[i] % 2 != 0 && arr[j] % 2 == 0)
            {
                swap(arr[i], arr[j]);
            }
            if(arr[i] % 2 == 0)
            {
                i += 2;
            }
            if(arr[j] % 2 != 0)
            {
                j += 2;
            }
        }
    }
}
```

};

12 [平均分19.4分 | 49人正确/63人做题 | 提交: 2 次 | 得分: 25.0 / 25.0

标题: 猴子分桃 | 时间限制: 1秒 | 内存限制: 32768K | 语言限制: 不限

【猴子分桃】老猴子辛苦了一辈子，给那群小猴子们留下了一笔巨大的财富——一大堆桃子。老猴子决定把这些桃子分给小猴子。

第一个猴子来了，它把桃子分成五堆，五堆一样多，但还多出一个。它把剩下的一个留给老猴子，自己拿走其中的一堆。

第二个猴子来了，它把桃子分成五堆，五堆一样多，但又多出一个。它把多出的一个留给老猴子，自己拿走其中的一堆。

后来的小猴子都如此照办。最后剩下的桃子全部留给老猴子。

这里有n只小猴子，请你写个程序计算一下在开始时至少有多少个桃子，以及最后老猴子最少能得到几个桃子。

输入描述：

输入包括多组测试数据。  
每组测试数据包括一个整数n(1≤n≤20)。  
输入以0结束，该行不做处理。

输出描述：

每组测试数据对应一行输出。  
包括两个整数a，b。  
分别代表开始时最小需要的桃子数，和结束后老猴子最少能得到的桃子数。

示例1：

输入

5  
1  
0

输出

3121 1025  
1 1

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 77%	使用语言	TA的 C++	平均	答案错误	: 1
基本测试用例通过率	1/1 (100%)	77%	做题用时	00:00:00	00:41:19	答案正确	: 1
			提交次数	2	4		
代码效率					代码规范及可读性		
运行时间	TA的 20ms	参考 1s				代码规范得分	5.0
占用内存	480K	32768K					

他的代码：

做题用时: <1 分钟    语言: C++    运行时间: 20ms    占用内存: 480K    程序状态: 答案正确

```
// write your code here cpp
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    //根据一大堆推导可以得出x = 5 ^ n - 4;
```

```
//给老猴子的桃子为:4 ^ n - 4 + n;
int n = 0;
while(cin >> n)
{
    if(n == 0)
    {
        break;
    }
    std::cout << std::fixed;
    std::cout.precision(0);
    cout << (pow(5, n) - 4) << " " << (pow(4, n) - 4 + n) << endl;
}
}
```