

# 每日一题day24\_6月15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：01:17:43(2019-06-16 00:00:26 - 01:18:14)

## 考生成绩



## 知识点技能图谱



## 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

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
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

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	00:13:49	C++	7ms	504K	良	良	1%
编程题2	100%	3	00:34:20	C++	3ms	492K	优	优	1%

1

[平均分1.5分 | 19人正确/65人做题 | 用时：5分] 得分：0.0 / 5.0

将N条长度均为M的有序链表进行合并，合并以后的链表也保持有序，时间复杂度为()？

A O(N \* M \* logN)

- B  $O(N \cdot M)$
- C  $O(N)$
- D  $O(M)$

他的回答： B (错误)

正确答案： A

2 [平均分4.4分 | 57人正确/65人做题 | 用时：<1分 | 得分：5.0 / 5.0

下设法S的初始状态为空，元素a,b,c,d,e,f依次入栈S，出栈的序列为b,d,c,f,e,a则栈S的容量至少为（ ）

- A 6
- B 5
- C 4
- D 3

他的回答： D (正确)

正确答案： D

3 [平均分2.9分 | 38人正确/65人做题 | 用时：<1分 | 得分：5.0 / 5.0

大小为MAX的循环队列中，f为当前对头元素位置，r为当前队尾元素位置(最后一个元素的位置)，则任意时刻，队列中的元素个数为

- A r-f
- B  $(r-f+MAX+1)\%MAX$
- C r-f+1
- D  $(r-f+MAX)\%MAX$

他的回答： B (正确)

正确答案： B

4 [平均分2.4分 | 29人正确/60人做题 | 用时：<1分 | 得分：5.0 / 5.0

n! 后面有多少个0,  $6! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 = 720$ . 720后面有1个0，n=10000，求n!。

- A 2498
- B 2499
- C 2450
- D 2451

他的回答： B (正确)

正确答案： B

5 [平均分3.3分 | 41人正确/62人做题 | 用时：2分 | 得分：5.0 / 5.0

若一棵二叉树具有12个度为2的结点，6个度为1的结点，则度为0的结点个数是（ ）。

- A 10
- B 11
- C 13
- D 不确定

他的回答： C (正确)

正确答案： C

6 [平均分2.5分 | 31人正确/61人做题 | 用时：8分 | 得分：5.0 / 5.0

若将关键字1, 2, 3, 4, 5, 6, 7 依次插入到初始为空的平衡二叉树T中，则T中平衡因子为0的分支结点的个数是（ ）。

- A 0
- B 1
- C 2
- D 3

他的回答：D (正确)

正确答案：D

7 [平均分2.0分 | 25人正确/64人做题 | 用时：<1分 | 得分：5.0 / 5.0

已知小根堆为8,15,10,21,34,16,12，删除关键字 8 之后需重建堆，在此过程中，关键字之间的比较次数是（ ）。

- A 1
- B 2
- C 3
- D 4

他的回答：C (正确)

正确答案：C

8 [平均分2.3分 | 25人正确/55人做题 | 用时：2分 | 得分：0.0 / 5.0

已知某个哈希表的n个关键字具有相同的哈希值，如果使用二次探测再散列法将这n个关键字存入哈希表，至少要进行\_\_\_\_\_次探测。

- A n-1
- B n
- C n+1
- D n(n+1)
- E n(n+1)/2
- F 1+n(n+1)/2

他的回答：B (错误)

正确答案：E

9 [平均分1.9分 | 22人正确/58人做题 | 用时：<1分 | 得分：0.0 / 5.0

下列选项中，不可能是快速排序第2趟排序结果的是（ ）

- A 2,3,5,4,6,7,9
- B 2,7,5,6,4,3,9
- C 3,2,5,4,7,6,9
- D 4,2,3,5,7,6,9

他的回答：D (错误)

正确答案：C

10 [平均分0.8分 | 9人正确/54人做题 | 用时：2分 | 得分：0.0 / 5.0

设有向图G=(V,E)，顶点集 V={V0,V1,V2,V3}，边集 E={<v0,v1>，<v0,v2>，<v0,v3>，<v1,v3>}。若从顶点 V0 开始对图进行深度优先遍历，则可能得到的不同遍历序列个数是（ ）。

- A 2
- B 3
- C 4
- D 5

他的回答：C (错误)

正确答案：D

标题：年终奖 | 时间限制：3秒 | 内存限制：32768K | 语言限制：[Python, C++, C#, Java]

【年终奖】

小东所在公司要发年终奖，而小东恰好获得了最高福利，他要在公司年会上参与一个抽奖游戏，游戏在一个6\*6的棋盘上进行，上面放着36个价值不等的礼物，每个小的棋盘上面放置着一个礼物，他需要从左上角开始游戏，每次只能向下或者向右移动一步，到达右下角停止，一路上的格子中的礼物小东都能拿到，请设计一个算法使小东拿到价值最高的礼物。

给定一个6\*6的矩阵**board**，其中每个元素为对应格子的礼物价值,左上角为[0,0],请返回能获得的最大价值，保证每个礼物价值大于100小于1000。

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	答案正确：1		
总通过率	100%	95%	使用语言	C++				
基本测试用例通过率	1/1 (100%)	95%	做题用时	00:13:49	00:29:36			
			提交次数	1	4			
代码效率					代码规范及可读性			
	TA的	参考					代码规范得分	4.6
运行时间	7ms	3s					Line 21: Add #include for max [build/include_what_you_use] [4]	
占用内存	504K	32768K					Line 5: Add #include for vector<> [build/include_what_you_use] [4]	

他的代码：

做题用时: 13 分钟    语言：C++    运行时间：7ms    占用内存：504K    程序状态：答案正确

```
class Bonus {
public:
    int getMost(vector<vector<int> > board) {
        // write code here
        vector<vector<int>> price(6, vector<int>(6, 0));
        price[0][0] = board[0][0];
        for(int i = 1; i < 6; i++)
        {
            price[0][i] = price[0][i - 1] + board[0][i];
        }
        for(int i = 1; i < 6; i++)
        {
            price[i][0] = price[i - 1][0] + board[i][0];
        }
        for(int i = 1; i < 6; i++)
        {
            for(int j = 1; j < 6; j++)
            {
                int num1 = price[i - 1][j] + board[i][j];
                int num2 = price[i][j - 1] + board[i][j];
                price[i][j] = max(num1, num2);
            }
        }
        return price[5][5];
    }
};
```

标题: 迷宫问题 | 时间限制: 1秒 | 内存限制: 32768K | 语言限制: 不限

【迷宫问题】

定义一个二维数组N\*M ( 其中 $2 \leq N \leq 10$ ;  $2 \leq M \leq 10$  ) , 如 $5 \times 5$ 数组下所示 :

```
int maze[5][5] = {  
  
    0, 1, 0, 0, 0,  
  
    0, 1, 0, 1, 0,  
  
    0, 0, 0, 0, 0,  
  
    0, 1, 1, 1, 0,  
  
    0, 0, 0, 1, 0,  
  
};
```

它表示一个迷宫, 其中的1表示墙壁, 0表示可以走的路, 只能横着走或竖着走, 不能斜着走, 要求程序找出从左上角到右下角的最短路线。入口点为[0,0],既第一空格是可以走的路。

Input

一个 $N \times M$ 的二维数组, 表示一个迷宫。数据保证有唯一解, 不考虑有多解的情况, 即迷宫只有一条通道。

Output

左上角到右下角的最短路径, 格式如样例所示。

Sample Input

```
0 1 0 0 0  
0 1 0 1 0  
0 0 0 0 0  
0 1 1 1 0  
0 0 0 1 0
```

Sample Output

```
(0, 0)  
(1, 0)  
(2, 0)  
(2, 1)  
(2, 2)  
(2, 3)  
(2, 4)  
(3, 4)  
(4, 4)
```

输入描述 :

输入两个整数, 分别表示二维数组的行数, 列数。再输入相应的数组, 其中的1表示墙壁, 0表示可以走的路。数据保证有唯一解, 不考虑有多解的情况, 即迷宫只有一条通道。

输出描述 :

左上角到右下角的最短路径, 格式如样例所示。

示例1 :

输入

```
5 5  
0 1 0 0 0  
0 1 0 1 0  
0 0 0 0 0  
0 1 1 1 0  
0 0 0 1 0
```

输出

```
(0,0)  
(1,0)  
(2,0)  
(2,1)  
(2,2)  
(2,3)
```

(2,4)  
(3,4)  
(4,4)

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	编译错误 : 2 答案正确 : 1	
总通过率	100%	71%	使用语言	C++			
基本测试用例通过率	6/6 (100%)	71%	做题用时	00:34:20	00:54:41		
边缘测试用例通过率	4/4 (100%)	72%	提交次数	3	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	492K	32768K					

他的代码：

做题用时: 34 分钟    语言：C++    运行时间：3ms    占用内存：492K    程序状态：答案正确

```
#include <iostream>
#include <vector>
using namespace std;
int m, n;
vector<vector<int>> maze;
vector<vector<int>> trackTemp;
vector<vector<int>> trackBest;
void MazeTrack(int i, int j)
{
    maze[i][j] = 1;
    trackTemp.push_back({i, j});
    if(i + 1 == m && j + 1 == n && (trackTemp.size() < trackBest.size() || trackBest.size() == 0))
    {
        trackBest = trackTemp;
    }
    if((i - 1 >= 0) && (maze[i - 1][j] == 0))
    {
        MazeTrack(i - 1, j);
    }
    if((i + 1 < m) && (maze[i + 1][j] == 0))
    {
        MazeTrack(i + 1, j);
    }
    if((j - 1 >= 0) && (maze[i][j - 1] == 0))
    {
        MazeTrack(i, j - 1);
    }
    if((j + 1 < n) && (maze[i][j + 1] == 0))
    {
        MazeTrack(i, j + 1);
    }
    maze[i][j] = 0;
    trackTemp.pop_back();
}
int main()
{
    while(cin >> m >> n)
```

```
{
    maze = vector<vector<int>>(m, vector<int>(n, 0));
    trackTemp.clear();
    trackBest.clear();
    for(int i = 0; i < m; i++)
    {
        for(int j = 0; j < n; j++)
        {
            cin >> maze[i][j];
        }
    }
    MazeTrack(0, 0);
    for(int i = 0; i < trackBest.size(); i++)
    {
        cout << "(" << trackBest[i][0] << "," << trackBest[i][1] << ")" << endl;;
    }
}
}
```