

# 每日一题day10\_5月30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：01:37:31(2019-05-29 22:24:35 - 2019-05-30 00:02:10)

## 考生成绩

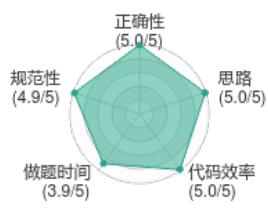


## 知识点技能图谱



## 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
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



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	00:14:27	C++	4ms	488K	优	优	1%
编程题2	100%	2	00:20:16	C++	4ms	476K	优	优	1%

1 [平均分2.5分 | 78人正确/154人做题 | 用时：<1分] 得分：0.0 / 5.0

32位系统中，定义\*\*a[3][4],则变量占用内存空间为()。

- A 4
- B 48
- C 192
- D 12

他的回答：A (错误)

正确答案：B

2 [平均分2.6分 | 77人正确/148人做题 | 用时：21分] 得分：5.0 / 5.0

二维数组X按行顺序存储，其中每个元素占1个存储单元。若X[4][4]的存储地址为0xf8b82140,X[9][9]的存储地址为0xf8b8221c,则X[7][7]的存储地址为()。

- A 0xf8b821c4
- B 0xf8b821a6
- C 0xf8b82198
- D 0xf8b821c0

他的回答：A (正确)

正确答案：A

3 [平均分3.4分 | 103人正确/151人做题 | 用时：6分] 得分：5.0 / 5.0

求函数返回值，输入x=9999

```
int func(int x){
    int count=0;
    while (x)
    {
        count++;
        x=x&(x-1);//与运算
    }
    return count;
}
```

- A 8
- B 9
- C 10
- D 12

他的回答：A (正确)

正确答案：A

4 [平均分4.6分 | 138人正确/150人做题 | 用时：<1分] 得分：5.0 / 5.0

根据下面递归函数：调用函数Fun（2），返回值是多少（）

```
int Fun(int n)
```

```
{
    if(n==5)
        return 2;
    else
        return 2*Fun(n+1);
}
```

- A 2
- B 4
- C 8
- D 16

他的回答： D (正确)

正确答案： D

5 [平均分3.1分 | 93人正确/150人做题 | 用时：2分  得分：5.0 / 5.0

执行下面语句后的输出为

```
int l=1;
if(l<=0)
    printf("****\n");
else
    printf("%%%%\n");
```

- A %%
- B \*\*\*\*
- C 有语法错，不能正确执行
- D %%%%

他的回答： A (正确)

正确答案： A

6 [平均分2.9分 | 81人正确/142人做题 | 用时：<1分  得分：5.0 / 5.0

在C++，下列哪一个可以做为对象继承之间的转换()

- A static\_cast
- B reinterpret\_cast
- C dynamic\_cast
- D const\_cast

他的回答： C (正确)

正确答案： C

7 [平均分2.8分 | 82人正确/149人做题 | 用时：<1分  得分：5.0 / 5.0

类模板的使用实际上是类模板实例化成一个具体的\_\_\_\_\_。

- A 类
- B 函数
- C 模板类
- D 对象

他的回答： A (正确)

正确答案： A

8 [平均分2.8分 | 82人正确/147人做题 | 用时：11分  得分：0.0 / 5.0

有如下C++代码：

```
struct A{
    void foo(){printf("foo");}
    virtual void bar(){printf("bar");}
    A(){bar();}
};
struct B:A{
    void foo(){printf("b_foo");}
    void bar(){printf("b_bar");}
};
```

那么

```
A *p=new B;
p->foo();
p->bar();
```

输出为：

- A barfoob\_bar
- B foobarb\_bar
- C barfoob\_foo
- D foobarb\_fpp

他的回答： B (错误)

正确答案： A

9 [平均分3.0分 | 90人正确/148人做题 | 用时：3分] 得分：5.0 / 5.0

下面的程序输出可能是什么？

```
class Printer{
public:
    Printer(std::string name) {std::cout << name;}
};
class Container{
public:
    Container() : b("b"), a("a") {}
    Printer a;
    Printer b;
};
int main(){
    Container c;
    return 0;
}
```

- A 可能是 "ab" 或 "ba"。依赖于具体的实现
- B 一直都是 "ba"
- C 一直都是 "ab"

他的回答： C (正确)

正确答案： C

10 [平均分2.1分 | 61人正确/142人做题 | 用时：11分] 得分：5.0 / 5.0

代码可以通过编译吗？如果不能应该如何修改？

```
template<class T> class Foo{
    T tVar;
public:
    Foo(T t) : tVar(t) { }
```

```
};

template<class T> class FooDerived:public Foo<T>
{
};

int main()
{
    FooDerived<int> d(5);
    return 0;
}
```

- A 代码可以正确通过编译。
- B 编译错误，FooDerived是一个继承模板类的非模板类，它的类型不能改变。
- C 编译错误，tVal变量是一个不确定的类型。
- D 编译错误，可以在FooDerived类中添加一个构造函数解决问题。

他的回答： D (正确)

正确答案： D

11 [平均分23.0分 | 118人正确/128人做题 | 提交: 3 次 | 得分：25.0 / 25.0]

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

【井字棋】

对于一个给定的井字棋棋盘，请设计一个高效算法判断当前玩家是否获胜。

给定一个二维数组board，代表当前棋盘，其中元素为1的代表是当前玩家的棋子，为0表示没有棋子，为-1代表是对方玩家的棋子。

测试样例：

```
[[1,0,1],[1,-1,-1],[1,-1,0]]
```

返回：true

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		100%	92%	使用语言		C++		编译错误：2	
基本测试用例通过率		1/1 (100%)	92%	做题用时		00:14:27	00:26:24	答案正确：1	
				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	
运行时间		4ms	3s					4.8	
占用内存		488K	32768K					Line 3: Add #include for vector<> [build/include_what_you_use]	
								[4]	

他的代码：

做题用时：14 分钟    语言：C++    运行时间：4ms    占用内存：488K    程序状态：答案正确

```
class Board {
public:
    bool checkWon(vector<vector<int> > board) {
```

```
// write code here
for(int i = 0; i < 3; i++)
{
    int sum = board[0][i] + board[1][i]+ board[2][i];
    if(sum == 3)
    {
        return true;
    }
}
for(int i = 0; i < 3; i++)
{
    int sum = board[i][0] + board[i][1]+ board[i][2];
    if(sum == 3)
    {
        return true;
    }
}
int sum = board[0][0] + board[1][1] + board[2][2];
if(sum == 3)
{
    return true;
}
sum = board[0][2] + board[1][1] + board[2][0];
if(sum == 3)
{
    return true;
}
return false;
}
};
```

12 [平均分18.4分 | 83人正确/113人做题 | 提交: 2 次] 得分 : 25.0 / 25.0

标题: 密码强度等级 | 时间限制: 1秒 | 内存限制: 32768K | 语言限制: 不限

#### 【密码强度等级】

密码按如下规则进行计分，并根据不同的得分为密码进行安全等级划分。

##### 一、密码长度:

5 分: 小于等于4 个字符

10 分: 5 到7 字符

25 分: 大于等于8 个字符

##### 二、字母:

0 分: 没有字母

10 分: 全都是小 ( 大 ) 写字母

20 分: 大小写混合字母

##### 三、数字:

0 分: 没有数字

10 分: 1 个数字

20 分: 大于1 个数字

##### 四、符号:

0 分: 没有符号

10 分: 1 个符号

25 分: 大于1 个符号

##### 五、奖励:

2 分: 字母和数字

3 分: 字母、数字和符号

5 分: 大小写字母、数字和符号

最后的评分标准:

>= 90: 非常安全

>= 80: 安全 ( Secure )

>= 70: 非常强

>= 60: 强 ( Strong )

>= 50: 一般 ( Average )

>= 25: 弱 ( Weak )

>= 0: 非常弱

对应输出为：

VERY\_WEAK,  
WEAK,  
AVERAGE,  
STRONG,  
VERY\_STRONG,  
SECURE,  
VERY\_SECURE

请根据输入的密码字符串，进行安全评定。  
注：  
字母：a-z, A-Z  
数字：0-9  
符号包含如下：(ASCII码表可以在UltraEdit的菜单view->ASCII Table查看)  
!"#\$%&'()\*+,-./ (ASCII码：x21~0x2F)  
:;<=>?@ (ASCII码：x3A~0x40)  
[\]^\_` (ASCII码：x5B~0x60)  
{|}~ (ASCII码：x7B~0x7E)

接口描述：

Input Param  
String pPasswordStr: 密码，以字符串方式存放。  
Return Value  
根据规则评定的安全等级。

```
public static Safelevel GetPwdSecurityLevel(String pPasswordStr)
{
    /*在这里实现功能*/
    return null;
}
```

输入描述：

输入一个string的密码

输出描述：

输出密码等级

示例1：

输入

38\$@NoNoNo

输出

VERY\_SECURE

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	答案错误：1 答案正确：1	
总通过率	100%	73%	使用语言	C++			
基本测试用例通过率	6/6 (100%)	73%	做题用时	00:20:16	00:57:27		
边缘测试用例通过率	4/4 (100%)	73%	提交次数	2	7		
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	4ms	1s					
占用内存	476K	32768K					

他的代码：

做题用时: 20 分钟    语言：C++    运行时间：4ms    占用内存：476K    程序状态：答案正确

```

#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str;
    while(cin >> str)
    {
        int grade = 0;
        int numOfBigAlpha = 0;
        int numOfSmallAlpha = 0;
        int numOfDigit = 0;
        int numOfOther = 0;
        if(str.size() <= 4)
        {
            grade += 5;
        }
        else if(str.size() >= 5 && str.size() <= 7)
        {
            grade += 10;
        }
        else if(str.size() >= 8)
        {
            grade += 25;
        }
        for(int i = 0; i < str.size(); i++)
        {
            if(str[i] >= 'a' && str[i] <= 'z')
            {
                numOfSmallAlpha++;
            }
            else if(str[i] >= 'A' && str[i] <= 'Z')
            {
                numOfBigAlpha++;
            }
            else if(str[i] >= '0' && str[i] <= '9')
            {
                numOfDigit++;
            }
            else
            {
                numOfOther++;
            }
        }
        if(numOfSmallAlpha == 0 || numOfBigAlpha == 0)
        {
            grade += 10;
        }
        else if(numOfSmallAlpha != 0 && numOfBigAlpha != 0)
        {
            grade += 25;
        }
        if(numOfDigit == 1)
        {
            grade += 10;
        }
        else if(numOfDigit > 1)
        {
            grade += 25;
        }
    }
}

```



```

    }
    if(numOfOther == 1)
    {
        grade += 10;
    }
    else if(numOfOther > 1)
    {
        grade += 25;
    }
    if(numOfDigit != 0 && ((numOfSmallAlpha + numOfBigAlpha) != 0))
    {
        grade += 2;
    }
    if(numOfDigit != 0 && ((numOfSmallAlpha + numOfBigAlpha) != 0) && numOfOther != 0)
    {
        grade += 3;
    }
    if(numOfDigit != 0 && numOfSmallAlpha != 0 && numOfBigAlpha != 0 && numOfOther != 0)
    {
        grade += 5;
    }
    if(grade >= 90)
    {
        cout << "VERY_SECURE" << endl;
    }
    else if(grade >= 80)
    {
        cout << "SECURE" << endl;
    }
    else if(grade >= 70)
    {
        cout << "VERY_STRONG" << endl;
    }
    else if(grade >= 60)
    {
        cout << "STRONG" << endl;
    }
    else if(grade >= 50)
    {
        cout << "AVERAGE" << endl;
    }
    else if(grade >= 25)
    {
        cout << "WEAK" << endl;
    }
    else if(grade >= 0)
    {
        cout << "VERY_WEAK" << endl;
    }
}
}

```