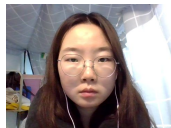


# C++方向每日一题day10\_4月29日-王梅-测评结果

## 考生信息



王梅

投递编号：62 | 学校：陕西科技大学 | 邮箱：2910602199@qq.com | 职位：91班 |

参考区域：陕西省西安市（123.139.160.69） |

做题用时：02:40:13(2021-04-28 21:04:05开始答题，2021-04-29 07:43:41交卷) | 作答设备：PC |

已同意诚信声明和隐私协议

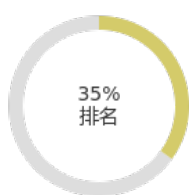
## 考生成绩



78.1  
得分



02:40:13  
用时



35%  
排名

题型	得分	正确题数	排名	用时	是否阅卷
单选	30.0	6	21	00:30:16	已阅
编程	48.1	1	19	02:05:11	已阅

## 知识点技能图谱

其他知识点	<div></div>	2/3
编程基础	<div></div>	1/1
C/C++	<div></div>	2/3
C++	<div></div>	2/5
数组	<div></div>	1/1

知识点	得分	正确题数
其他知识点	33.1	2
编程基础	25.0	1
C/C++	10.0	2
C++	10.0	2
数组	25.0	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	73班C1考试试卷	44.0%	22.0/60	单选:22.0分 编程:0.0分	否	2020-05-17 15:06:33	2020-05-17 23:09:23
2	实习生冲刺班C2考试试卷	19.0%	37.5/60	单选:18.0分 编程:19.5分	否	2020-07-14 10:30:11	2020-07-18 22:56:30
3	C++方向每日一题day01_4月19日	19.0%	88.8/100	单选:45.0分 编程:43.75分	否	2021-04-18 11:21:15	2021-04-19 19:27:23
4	C++方向每日一题day02_4月20日	56.0%	57.5/100	单选:25.0分 编程:32.5分	否	2021-04-19 14:53:21	2021-04-20 20:22:40
5	C++方向每日一题day03_4月21日	7.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-04-20 11:16:00	2021-04-21 16:09:24

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
6	C++方向每日一题day04_4月22日	4.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-04-21 16:19:39	2021-04-22 16:12:03
7	C++方向每日一题day05_4月23日	1.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-04-22 15:09:26	2021-04-23 10:24:19
8	C++方向每日一题day06_4月24日	6.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-04-22 15:21:12	2021-04-24 17:35:03
9	C++方向每日一题day07_4月26日	1.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-04-22 16:20:07	2021-04-26 18:10:43
10	C++方向每日一题day08_4月27日	6.0%	95.0/100	单选:50.0分 编程:45.0分	否	2021-04-24 15:13:08	2021-04-27 15:21:45
11	91班&92班CPP1考试试卷	26.0%	68.0/80	单选:18.0分 编程:50.0分	否	2021-04-26 14:46:52	2021-04-28 20:07:07
12	C++方向每日一题day09_4月28日	18.0%	90.0/100	单选:40.0分 编程:50.0分	否	2021-04-27 16:33:22	2021-04-28 14:46:16

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%	7	00:35:49	C++	3ms	448K			1%
编程题2	92%	16	01:29:22	C++	4ms	380K			35%

1

[平均分2.3分 | 21人正确/46人做题 | 用时：<1分]

得分：5.0 / 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： B (正确)

正确答案： B

2

[平均分3.1分 | 27人正确/44人做题 | 用时：3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Oxf8b82140,X[9][9]的存储地址为Oxf8b8221c,则X[7][7]的存储地址为()。

A Oxf8b821c4  
B Oxf8b821a6  
C Oxf8b82198  
D Oxf8b821c0

他的回答： A (正确)

正确答案： A

3 [平均分3.8分 | 34人正确/45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 | 41人正确/45人做题 | 用时：2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3.5分 | 32人正确/46人做题 | 用时：<1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2.3分 | 21人正确/46人做题 | 用时：2分 得分：0.0 / 5.0

在C++，下列哪一个可以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：B (错误)

正确答案：C

7 [平均分1.8分 | 17人正确/47人做题 | 用时：<1分 得分：0.0 / 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：D (错误)

正确答案：A

8 [平均分3.2分 | 29人正确/46人做题 | 用时：4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 | 28人正确/46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2.2分 | 21人正确/47人做题 | 用时：3分 | 得分：0.0 / 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： B (错误)

正确答案： D

11 完善核心代码 语言限制 [平均分20.0分 | 36人正确/45人做题 | 提交: 7 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	答案错误：2	
总通过率		100%	80%	使用语言		C++		编译错误：3	
基本测试用例通过率		1/1 (100%)	80%	做题用时		00:35:49 00:17:46		答案正确：2	
				提交次数		7 2			
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	3s				4.77778		
占用内存		448K	32768K				Line 3: Add #include for vector<> [build/include_what_you_use] [4]		

他的代码：

做题用时: 35 分钟    语言: C++    运行时间: 3ms    占用内存: 448K    程序状态: 答案正确

```
class Board {
public:
    bool checkWon(vector<vector<int> > board) {
        for(int i = 0; i < 3; i++)
        {
            int count1 = 0;
            int count2 = 0;
            for(int j = 0; j < 3; j++)
            {
                if(board[i][j] == 1)
                {
                    count1++;
                    if(count1 == 3)
                    {
                        return true;
                    }
                }
            }
            if(board[j][i] == 1)
            {
                count2++;
                if(count2 == 3)
                {
                    return true;
                }
            }
        }
        int count3 = 0;
        for(int i = 0; i < 3; i++)
        {
            for(int j = 0; j < 3; j++)
            {
                if((i == j || (3-i) == j) && (board[i][j] == 1))
                {
                    count3++;
                }
            }
        }
        if(count3 == 3)
        {
            return true;
        }
        return false;
    }
};
```

```
class Board {
public:
    bool checkWon(vector<vector<int> > board) {
        int row = (board.size());
        //检查每一行的和是否等于row
        int i, j, sum;
        for(i = 0; i < row; i++)
        {
            sum = 0;
            for(j = 0; j < row; j++)
            {
                sum += board[i][j];
            }
            if(sum == row)
            {
                return true;
            }
        }
        //检查每一列的和是否等于row
        for(i = 0; i < row; i++)
        {
            sum = 0;
            for(j = 0; j < row; j++)
            {
                sum += board[j][i];
            }
            if(sum == row)
            {
                return true;
            }
        }
        //检查对角线的和是否等于row
        sum = 0;
        for(i = 0; i < row; i++)
        {
            sum += board[i][i];
        }
        if(sum == row)
        {
            return true;
        }
        //检查副对角线的和是否等于row
        sum = 0;
        for(i = 0; i < row; i++)
        {
            sum += board[i][row-i-1];
        }
        if(sum == row)
        {
            return true;
        }
        return false;
    }
};
```

```
        if(count3 == 3)
        {
            return true;
        }
    }
}
return false;
};
```



[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 [平均分14.7分 | 18人正确/44人做题 | 提交: 16 次] 得分 : 23.1 / 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：-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		
总通过率	TA的 92%	平均 58%	使用语言	TA的 C++	做题用时	平均 01:29:22	00:52:14	答案错误：16	
基本测试用例通过率	8/8 (100%)	61%	提交次数	16	6				
边缘测试用例通过率	4/5 (80%)	54%							
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	380K	32768K							

他的代码：
 做题用时: 89 分钟
 语言：C++
 运行时间：4ms
 占用内存：380K
 程序状态：答案错误

```

#include<iostream>
#include<string>
using namespace std;
int count(string str)
{

```



```

int len = str.length();
int grade = 0;
if (len <= 4 && len >= 0)
    grade = 5;
if (len >= 5 && len <= 7)
    grade = 10;
if (len >= 8)
    grade = 25;
int num = 0;
int Bletter = 0;
int Sletter = 0;
int letter = 0;
int sign = 0;
int add = 0;
for (int i = 0; i < len; i++)
{
    if (str[i] >= 'a' && str[i] <= 'z')
    {
        Sletter = 1;
    }
    if (str[i] >= 'A' && str[i] <= 'Z')
    {
        Bletter = 1;
    }
    if (Sletter == 1 && Bletter == 1)
    {
        letter = grade + 25;
    }
    if (str[i] >= '0' && str[i] <= '9')
    {
        num++;
    }
    if (str[i] < '0' || (str[i] > '9' && str[i] < 'A') || (str[i] > 'Z' && str[i] < 'a') || str[i] > 'z')
    {
        sign++;
    }
    if ((Bletter == 1 || Sletter == 1) && num != 0)
    {
        add = 2;
        if (sign != 0)
        {
            add = 3;
        }
        if (letter != 0)
        {
            add = 5;
        }
    }
}

if(Sletter == 1 || Bletter == 1)
    grade += 10;
if (letter != 0)
    grade = letter;
if (num == 0)
    grade += 0;
if (num == 1)
    grade += 10;
if (num > 1)
    grade += 20;
if (sign == 0)

```

```

grade += 0;
if (sign == 1)
    grade += 10;
if (sign > 1)
    grade += 25;
if (add != 0)
    grade += add;
return grade;
}

int main()
{
    string str;
    while(cin >> str)
    {
        int ret = count(str);
        if (ret >= 90)
        {
            cout << "VERY_SECURE" << endl;
        }
        if (ret >= 80 && ret < 90)
        {
            cout << "SECURE" << endl;
        }
        if (ret >= 70 && ret < 80)
        {
            cout << "VERY_STRONG" << endl;
        }
        if (ret >= 60 && ret < 70)
        {
            cout << "STRONG" << endl;
        }
        if (ret >= 50 && ret < 60)
        {
            cout << "AVERAGE" << endl;
        }
        if (ret >= 25 && ret < 50)
        {
            cout << "WEAK" << endl;
        }
        if (ret >= 0 && ret < 25)
        {
            cout << "VERY_WEAK" << endl;
        }
    }
    return 0;
}

```



```
#include<bits/stdc++.h>
using namespace std;
string str;
int main()
{
    while(cin >> str)
    {
        int n = str.size();
        int numn = 0, charc = 0, grade = 0;
        bool word[2] = {false, false};
        if( n<= 4)
            grade += 5;
        else if(n <= 7)
            grade += 10;
        else
            grade += 25;
        for(int i = 0; i < n; i++)
        {
            if(str[i] >= '0' && str[i] <= '9')
                numn++;
            if(str[i] >= 'a' && str[i] <= 'z')
                word[0] = true;
            if(str[i] >= 'A' && str[i] <= 'Z')
                word[1] = true;
            if((str[i] >= 0x21 && str[i] <= 0x2f) || (str[i] >= 0x3a && str[i] <= 0x40)
                || (str[i] >= 0x5b && str[i] <= 0x60) || (str[i] >= 0x7b && str[i] <= 0x7e))
                charc++;
        }
        if((word[0] == false && word[1] == true) || (word[1] == false && word[0] == true))
            grade += 10;
        else if(word[0] && word[1])
            grade += 20;
        if(numn == 1)
            grade += 10;
        else if(numn > 1)
            grade += 20;
        if((word[0] || word[1]) && numn)
        {
            if(charc == 0)
                grade += 2;
            if(word[0] && word[1])
                grade += 5;
            else
                grade += 3;
        }
        if(charc == 1)
            grade += 10;
        else if(charc > 1)
            grade += 25;
        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
            cout << "VERY_WEAK" << endl;
    }
    return 0;
}
```