

# C++方向每日一题day17\_5月14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（123.139.161.98） | 做题用时：02:22:48(2021-05-14 12:15:58开始答题，17:42:58交卷) |

作答设备：PC | 已同意诚信声明和隐私协议

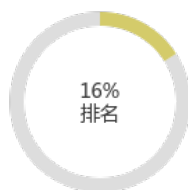
## 考生成绩



90.00  
得分



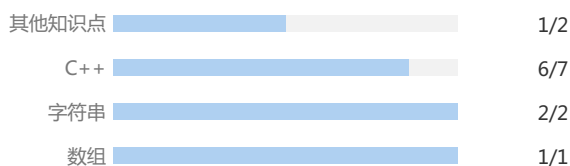
02:22:48  
用时



16%  
排名

题型	得分	正确题数	排名	用时	是否阅卷
单选	40.00	8	12	00:36:00	已阅
编程	50.00	2	1	01:43:56	已阅

## 知识点技能图谱



知识点	得分	正确题数
其他知识点	5.00	1
C++	30.00	6
字符串	50.00	2
数组	5.00	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
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

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
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
13	C++方向每日一题day10_4月29日	35.0%	78.1/100	单选:30.0分 编程:48.08分	否	2021-04-28 11:39:52	2021-04-29 07:43:41
14	C++方向每日一题day11_5月7日	1.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-05-06 15:16:10	2021-05-06 22:42:17
15	C++方向每日一题day12_5月8日	6.0%	95.0/100	单选:45.0分 编程:50.0分	否	2021-05-07 14:24:30	2021-05-08 12:57:48
16	C++方向每日一题day13_5月10日	1.0%	75.0/100	单选:50.0分 编程:25.0分	否	2021-05-09 11:22:20	2021-05-10 08:10:17
17	C++方向每日一题day14_5月11日	23.0%	70.0/100	单选:40.0分 编程:30.0分	否	2021-05-10 10:13:17	2021-05-10 22:06:18
18	C++方向每日一题day15_5月12日	8.0%	85.0/100	单选:35.0分 编程:50.0分	否	2021-05-10 14:52:33	2021-05-12 18:58:48
19	C++方向每日一题day16_5月13日	9.0%	82.9/100	单选:35.0分 编程:47.92分	否	2021-05-12 10:20:46	2021-05-12 23:22:37
20	91&92班Linux1考试试卷	21.0%	52.0/60	单选:18.0分 不定项选择:4.0分 编程:30.0分	否	2021-05-12 14:37:21	2021-05-15 09:44:02

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	00:16:51	C++	6ms	632K			1%
编程题2	100%	10	01:27:05	C++	6ms	396K			1%

1

[平均分3.78分 | 37人正确/49人做题 | 用时：2分]

得分：0.00 / 5.00

以下代码

```
char *ptr;
```

```
char myString[] = "abcdefg";  
ptr = myString;  
ptr += 5;
```

代码执行之后ptr指向的内容是？

- A Compiler error
- B f
- C efg
- D defg

他的回答： C (错误)

正确答案： B

2 [平均分4.39分 | 43人正确/49人做题 | 用时：2分] 得分：5.00 / 5.00

已知int a[]={1,2,3,4,5}; int\*p[]={a,a+1,a+2,a+3}; int \*\*q=p; 表达式\*(p[0]+1)+\*\*(q+2)的值是\_\_\_\_\_。

- A 5
- B 6
- C 7
- D 8
- E 4
- F 9

他的回答： A (正确)

正确答案： A

3 [平均分4.18分 | 41人正确/49人做题 | 用时：2分] 得分：5.00 / 5.00

以下代码的输出结果是？

```
char *p="abc";  
char *q="abc123";  
while(*p==*q)  
print("%c %c",*p,*q);
```

- A aabbcc
- B aabbcc123
- C abcabc123
- D 代码段错误

他的回答： D (正确)

正确答案： D

4 [平均分2.55分 | 25人正确/49人做题 | 用时：2分] 得分：0.00 / 5.00

假设在一个 32 位 little endian 的机器上运行下面的程序，结果是多少？

```
#include <stdio.h>  
int main(){  
    long long a = 1, b = 2, c = 3;  
    printf("%d %d %d\n", a, b, c);  
    return 0;  
}
```

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

他的回答： A (错误)

正确答案： B

5 [平均分3.02分 | 29人正确/48人做题 | 用时：7分] 得分：5.00 / 5.00

下列给定程序中，函数fun的功能是：求ss所指字符串数组中长度最短的字符串所在的行下标，作为函数值返回，并把其串长放在形参n所指的变量中。ss所指字符串数组中共有M个字符串，且串长小于N。

请在程序的下画线处填入正确的内容并将下画线删除，使程序得出正确的结果。

试题程序。

```
#define M 5
#define N 20
int fun(char(* ss)[N], int *n)
{
    int i, k = 0, len = N;
    for (i = 0; i < ____; i++)
    {
        len = strlen(ss[i]);
        if (i == 0)
            *n = len;
        if (len ____ * n)
        {
            *n = len;
            k = i;
        }
    }
    return ( ____ );
}
main()
{
    char ss[M][N] = {"shanghai", "guangzhou", "beijing", "tianjing", "chongqing"};
    int n, k, i;
    printf("\nThe original strings are:\n");
    for (i = 0; i < M; i++)
        puts(ss[i]);
    k = fun(ss, &n);
    printf("\nThe length of shortest string is: %d\n", n);
    printf("\nThe shortest string is: %s\n", ss[k]);
}
```

A N , < , k

B N , > , k

C M , < , k

D M , > , k

他的回答： C (正确)

正确答案： C

6 [平均分4.04分 | 38人正确/47人做题 | 用时：<1分] 得分：5.00 / 5.00

调用一成员函数时，使用动态联编的情况是？

A 通过对象调用一虚函数

B 通过指针或引用调用一虚函数

C 通过对象调用静态函数

D 通过指针或应用调用一静态函数

他的回答： B (正确)

正确答案：B

7 [平均分1.91分 | 18人正确/47人做题 | 用时：4分] 得分：5.00 / 5.00

如何捕获异常可以使得代码通过编译？

```
class A {  
    public:  
        A(){  
};  
void foo(){  
    throw new A;  
}
```

- A catch (A && x)
- B catch (A \* x)
- C catch (A & x)
- D 以上都是

他的回答：B (正确)

正确答案：B

8 [平均分3.40分 | 34人正确/50人做题 | 用时：3分] 得分：5.00 / 5.00

下列代码可以通过编译吗？如何修改使其通过编译？

```
template <class T>  
struct sum {  
    static void foo(T op1 , T op2){  
        cout << op1 << op2;  
    }  
};  
sum::foo(1,3);
```

- A 编译通过
- B 应该去掉static关键字
- C 调用应该如下：sum<int>:: foo(1,3)
- D 调用应该如下：sum:: <int>foo(1,3)

他的回答：C (正确)

正确答案：C

9 [平均分4.18分 | 41人正确/49人做题 | 用时：4分] 得分：5.00 / 5.00

下面这段程序的输出是什么？

```
class A{  
    public:  
        A(){p0;}  
        virtual void p0{print("A")}  
        virtual ~A(){p0;}  
};  
class B:public A{  
    public:  
        B(){p0;}  
        void p0{print("B")}  
        ~B(){p0;}  
};  
int main(int, char**){
```

```
A* a=new B();
delete a;
}
```

- A AABBB
- B BBAAA
- C ABAB
- D ABBA

他的回答： D (正确)

正确答案： D

10 [平均分2.14分 | 21人正确/49人做题 | 用时：4分] 得分：5.00 / 5.00

有如下程序段：

```
#include <iostream>
using namespace std;

class A {
public:
    ~A() {
        cout << "~A()";
    }
};

class B{
public:
    virtual ~B() {
        cout << "~B()";
    }
};

class C: public A, public B {
public:
    ~C() {
        cout << "~C()";
    }
};

int main() {
    C * c = new C;
    B * b1 = dynamic_cast<B *>(c);
    A * a2 = dynamic_cast<A *>(b1);
    delete a2;
}
```

则程序输出：

- A ~C()~B()~A()
- B ~C()~A()~B()
- C A)B)都有可能
- D 以上都不对

他的回答： D (正确)

正确答案： D

11 ACM编程题 语言限制 [平均分17.99分 | 33人正确/46人做题 | 提交: 1 次] 得分：25.00 / 25.00

标题：iNOC产品部-杨辉三角的变形 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【iNOC产品部-杨辉三角的变形】

1 1 1  
1 2 3 2 1  
1 3 6 7 6 3 1  
1 4 10 16 19 16 10 4 1

以上三角形的数阵，第一行只有一个数1，以下每行的每个数，是恰好是它上面的数，左上角数到右上角的数，3个数之和（如果不存在某个数，认为该数就是0）。求第n行第一个偶数出现的位置。如果没有偶数，则输出-1。例如输入3,则输出2，输入4则输出3。

输入n(n <= 1000000000)

输入描述：

输入一个int整数

输出描述：

输出返回的int值

示例1：

输入

4

输出

3

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%	71%	使用语言	C++				
基本测试用例通过率	6/6 (100%)	71%	做题用时	00:16:51	00:40:47			
			提交次数	1	3			
边缘测试用例通过率	4/4 (100%)	72%						
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	6ms	1s						
占用内存	632K	32768K						

他的代码：

做题用时: 16 分钟    语言: C++    运行时间: 6ms    占用内存: 632K    程序状态: 答案正确

```
#include<iostream>
using namespace std;
#include <deque>
void getPos(deque<int> s, int count)
{
    count = count - 1;
    s.push_front(0);
    s.push_front(0);
    s.push_back(0);
    s.push_back(0);
    deque<int> temp;
    for (int i = 1; i < static_cast<int>(s.size()) - 1; i++)
    {
        int t = s.at(i - 1) + s.at(i) + s.at(i + 1);
        if (count <= 0)
        {
            if (t % 2 == 0)
```

```

{
    cout << i << endl;
    return;
}
}
temp.push_back(t);
}
if (count <= 0)
{
    cout << -1 << endl;
    return;
}
s.clear();
s.insert(s.begin(), temp.begin(), temp.end());
temp.clear();
getPos(s, count);
}

int main()
{
    deque<int> s;
    s.push_back(1);
    int n;
    while (cin >> n)
    {
        if (n == 1)
            cout << -1;
        else
        {
            --n;
            getPos(s, n);
        }
    }
    return 0;
}

```



[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19.02分 | 35人正确/46人做题 | 提交: 10 次] 得分 : 25.00 / 25.00

标题 : 超长正整数相加 | 时间限制 : 1秒 | 内存限制 : 32768K | 语言限制 : 不限

【超长正整数相加】

请设计一个算法完成两个超长正整数的加法。

接口说明

/\*

请设计一个算法完成两个超长正整数的加法。



String addend : 加数  
String augend : 被加数  
返回值 : 加法结果

输入描述：

输出描述：

示例1：

[illegible]

1000

	TA的	平均
总通过率	100%	76%
基本测试用例通过率	6/6 (100%)	76%
边缘测试用例通过率	4/4 (100%)	76%

	TA的	平均
使用语言	C++	
做题用时	01:27:05	00:26:13
提交次数	10	3

答案错误 : 5  
编译错误 : 4  
答案正确 : 1

	TA的	参考
运行时间	6ms	1s
占用内存	396K	32768

代码规范得分	4.79592
Line 22: Using C-style cast. Use static_cast(...) instead [readability/casting] [4]	

做题用时: 87 分钟      语言: C++      运行时间: 6ms      占用内存: 396K      程序状态: 答案正确

```
#include<iostream>
using namespace std;
#include<string>
string add(string str1, string str2)
{
    //reverse(str1.begin(), str1.end());
    //reverse(str2.begin(), str2.end());
    string str;
    int cur = 0;
    int m = str1.size() - 1;
    int n = str2.size() - 1;
```

```
while (m >= 0 || n >= 0)
{
    if (m >= 0)
    {
        cur += (str1[m] - '0');
    }
    if (n >= 0)
    {
        cur += (str2[n] - '0');
    }
    str += (char)(cur % 10 + '0');
    cur /= 10;
    m--;
    n--;
}
if (cur == 1)
{
    str += '1';
}
return str;
}
int main()
{
    string str1, str2;
    while (getline(cin, str1), getline(cin, str2))
    {
        string str = add(str1, str2);
        string::reverse_iterator it = str.rbegin();
        while (it != str.rend())
        {
            cout << *it;
            ++it;
        }
        cout << endl;
    }
    add(str1, str2);
    return 0;
}
```



[点此](#)或手机扫描二维码查看代码编写过程