

C++方向每日一题day16_5月13日-王梅-测评结果

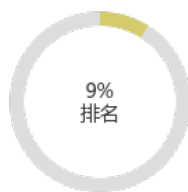
考生信息



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（123.139.170.134） | 做题用时：02:04:10(2021-05-12 19:01:16开始答题，23:22:37交卷) |
作答设备：PC | 已同意诚信声明和隐私协议

考生成绩



题型	得分	正确题数	排名	用时	是否阅卷
单选	35.00	7	9	00:40:31	已阅
编程	47.92	1	17	01:18:14	已阅

知识点技能图谱

其他知识点	2/2
C/C++	0/1
C++	5/7
字符串	0/1
链表	0/1
栈	0/1
数学	1/1
队列	0/1

知识点	得分	正确题数
其他知识点	10.00	2
C/C++	0.00	0
C++	25.00	5
字符串	22.92	0
链表	22.92	0
栈	22.92	0
数学	25.00	1
队列	22.92	0

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

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
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
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

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	00:11:35	C++	49ms	424K			1%
编程题2	92%	3	01:06:39	C++	2ms	428K			26%

1

[平均分1.11分 | 12人正确/54人做题 | 用时：5分]

得分：0.00 / 5.00

```
void swap_int(int *a,int *b){
```

```
    *a=*a+*b;  
    *b=*a-*b;  
    *a=*a-*b;  
}
```

以下说法正确的是：

- A 结果不正确，因为会溢出，用位与的方式就没问题
- B 结果正确，即使会溢出
- C 结果正确，不会溢出
- D 其他选项都不对

他的回答：A (错误)

正确答案：B

2 [平均分2.55分 | 28人正确/55人做题 | 用时：2分  得分：0.00 / 5.00

若有定义int (*pt) [3]；则下列说法正确的是：

- A 定义了基类型为int的三个指针变量
- B 定义了基类型为int的具有三个元素的指针数组pt
- C 定义了一个名为*pt、具有三个元素的整型数组
- D 定义了一个名为pt的指针变量，它可以指向每行有三个整数元素的二维数组

他的回答：C (错误)

正确答案：D

3 [平均分2.31分 | 24人正确/52人做题 | 用时：7分  得分：5.00 / 5.00

对于下面的说法，正确的是_____。

- A 对于 struct X { short s; int i; char c; }，sizeof(X) 的值等于 sizeof(s) + sizeof(i) + sizeof(c)
- B 对于某个double变量 a，可以使用 a == 0.0 来判断其是否为零
- C 初始化方式 char a[14] = "Hello, world!"; 和初始化方式 char a[14]; a = "Hello, world!"; 的效果相同
- D 在gcc编译器下，对于 int i = 3; printf("%d %d", ++i, ++i)，运行输出为：4 5
- E 选项A、B、C、D中至少有两个是正确的
- F 以上选项均不正确

他的回答：F (正确)

正确答案：F

4 [平均分2.83分 | 30人正确/53人做题 | 用时：3分  得分：5.00 / 5.00

下列代码的输出是？（注：print已经声明过）

```
main(){  
    char str[]="Geneius";  
    print (str);  
}  
print(char *s){  
    if(*s){  
        print(++s);  
        printf("%c",*s);  
    }  
}
```

- A suiene
- B neius

C run-time error
D suieneG

他的回答： A (正确)

正确答案： A

5 [平均分3.91分 | 43人正确/55人做题 | 用时：8分] 得分：5.00 / 5.00

写出下面程序的输出结果

```
class A
{
public:
void FuncA()
{
    printf( "FuncA called\n" );
}
virtual void FuncB()
{
    printf( "FuncB called\n" );
}
};

class B : public A
{
public:
void FuncA()
{
    A::FuncA();
    printf( "FuncAB called\n" );
}
virtual void FuncB()
{
    printf( "FuncBB called\n" );
}
};

void main( void )
{
    B b;
    A *pa;
    pa = &b;
    A *pa2 = new A;
    pa->FuncA(); ( 3 )
    pa->FuncB(); ( 4 )
    pa2->FuncA(); ( 5 )
    pa2->FuncB();
    delete pa2;
}
```

A FuncA called FuncB called FuncA called FuncB called
B FuncA called FuncBB called FuncA called FuncB called
C FuncA called FuncBB called FuncAB called FuncBB called
D FuncAB called FuncBB called FuncA called FuncB called

他的回答： B (正确)

正确答案： B

6 [平均分2.92分 | 31人正确/53人做题 | 用时：<1分] 得分：5.00 / 5.00

以下程序输出是_____。

```
#include <iostream>
using namespace std;
int main(void)
{
    const int a = 10;
    int * p = (int *)(&a);
    *p = 20;
    cout<<"a = "<<a<<" , *p = "<<*p<<endl;
    return 0;
}
```

A 编译阶段报错运行阶段报错

B a = 10, *p = 10

C a = 20, *p = 20

D a = 10, *p = 20

E a = 20, *p = 10

他的回答： D (正确)

正确答案： D

7 [平均分2.83分 | 30人正确/53人做题 | 用时：2分] 得分：5.00 / 5.00

以下关于STL的描述中，_____是错的。

A STL容器是线程不安全的

B 当容量不够时，vector内部内存扩展方式是翻倍

C std::sort是稳定排序

D std::bitset不是一个STL容器

E std::stack默认是用deque实现的

F std::string中可以存储多个' \0' 字符

他的回答： C (正确)

正确答案： C

参考答案：

要是能在这里留下这道题的解题思路，就再好不过啦

8 [平均分2.55分 | 27人正确/53人做题 | 用时：<1分] 得分：0.00 / 5.00

以下代码共调用多少次拷贝构造函数：

```
Widget f(Widget u)
{
    Widget v(u);
    Widget w=v;
    return w;
}
main(){
    Widget x;
    Widget y=f(f(x));
}
```

A 1

B 3

C 5

D 7

他的回答： B (错误)

正确答案： D

9 [平均分1.94分 | 21人正确/54人做题 | 用时：2分] 得分：5.00 / 5.00

以下代码有什么问题？

```
struct Test
{
    Test( int ) {}
    Test() {}
    void fun() {}
};
void main( void )
{
    Test a(1);
    a.fun();
    Test b();
    b.fun();
}
```

- A b.fun () 会出错
- B Test结构的定义中应该加上public修饰符，这样才能main函数中调用改类的方法
- C Test(int){} 应该改成Test(int a){}
- D 以上说法都不正确

他的回答： A (正确)

正确答案： A

10 [平均分2.78分 | 30人正确/54人做题 | 用时：5分] 得分：5.00 / 5.00

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

class Base
{
public:
    virtual int foo(int x)
    {
        return x * 10;
    }

    int foo(char x[14])
    {
        return sizeof(x) + 10;
    }
};

class Derived: public Base
{
    int foo(int x)
    {
        return x * 20;
    }

    virtual int foo(char x[10])
    {
        return sizeof(x) + 20;
    }
}
```

```
};

int main()
{
    Derived stDerived;
    Base *pstBase = &stDerived;

    char x[10];
    printf("%d\n", pstBase->foo(100) + pstBase->foo(x));

    return 0;
}
```

在32位环境下，以上程序的输出结果是？

- A 2000
- B 2004
- C 2014
- D 2024

他的回答： C (正确)

正确答案： C

11 ACM编程题 语言限制 [平均分22.61分 | 42人正确/47人做题 | 提交: 1 次 得分: 25.00 / 25.00]

标题：iNOC产品部--完全数计算 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【iNOC产品部--完全数计算】

完全数（Perfect number），又称完美数或完备数，是一些特殊的自然数。

它所有的真因子（即除了自身以外的约数）的和（即因子函数），恰好等于它本身。

例如：28，它有约数1、2、4、7、14、28，除去它本身28外，其余5个数相加， $1+2+4+7+14=28$ 。

给定函数count(int n),用于计算n以内(含n)完全数的个数。计算范围, $0 < n \leq 500000$

返回n以内完全数的个数。异常情况返回-1

```
/**
 *
 * 完全数（Perfect number），又称完美数或完备数，是一些特殊的自然数。
 * 它所有的真因子（即除了自身以外的约数）的和（即因子函数），恰好等于它本身。
 * 例如：28，它有约数1、2、4、7、14、28，除去它本身28外，其余5个数相加， $1+2+4+7+14=28$ 。
 *
 * 给定函数count(int n),用于计算n以内(含n)完全数的个数
 * @param n 计算范围,  $0 < n \leq 500000$ 
 * @return n 以内完全数的个数, 异常情况返回-1
 */
public static int count( int n)
```

输入描述：

输入一个数字

输出描述：

输出完全数的个数

示例1：

输入

1000

输出

3

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 90%	使用语言	TA的 C++	平均	答案正确 : 1	
基本测试用例通过率	6/6 (100%)	90%	做题用时	00:11:35	00:28:16		
边缘测试用例通过率	4/4 (100%)	90%	提交次数	1	4		
代码效率					代码规范及可读性		
运行时间	TA的 49ms	参考 1s	代码规范得分 5.0				
占用内存	424K	32768K					

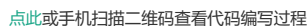
他的代码：

做题用时: 11 分钟 语言：C++ 运行时间：49ms 占用内存：424K 程序状态：答案正确

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

bool completeNum(int n)
{
    int sum = 0;
    for (int i = 1; i < n; i++)
    {
        if (n%i == 0)
        {
            sum += i;
        }
    }
    if (sum == n)
        return true;
    return false;
}

int main()
{
    int n;
    while (cin >> n)
    {
        int count = 0;
        for (int i = 1; i < n; i++)
        {
            if (completeNum(i) == 1)
            {
                count++;
            }
        }
        cout << count << endl;
    }
    return 0;
}
```

标题：扑克牌大小 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

扑克牌游戏大家应该都比较熟悉了，一副牌由54张组成，含3~A、2各4张，小王1张，大王1张。牌面从小到大用如下字符和字符串表示（其中，小写joker表示小王，大写JOKER表示大王）：

输入两手牌，两手牌之间用“-”连接，每手牌的每张牌以空格分隔，“-”两边没有空格，如：4 4 4 4-joker JOKER。

基本规则：

(2) 除了炸弹和对王可以和所有牌比较之外，其他类型的牌只能跟相同类型的存在比较关系（如，对子跟对子比较，三个跟三个比较），不考虑拆牌情况（如：将对子拆分成个子）；

(4) 输入的两手牌不会出现相等的情况。

输入两手牌，两手牌之间用“-”连接，每手牌的每张牌以空格分隔，“-”两边没有空格，如 4 4 4 4-joker JOKER。

输出两手牌中较大的那手，不含连接符，扑克牌顺序不变，仍以空格隔开；如果不存在比较关系则输出ERROR。

输入

4 4 4 4-joker JOKER

joker JOKER

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 : 3
总通过率	92%	45%	使用语言	C++		
基本测试用例通过率	6/7 (86%)	45%	做题用时	01:06:39	00:47:26	
边缘测试用例通过率	5/5 (100%)	46%	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	2ms	1s				
占用内存	428K	32768K				

他的代码：

做题用时: 66 分钟

语言: C++

运行时间: 2ms

占用内存: 428K

程序状态: 答案错误

```
#include<iostream>
#include<string>
using namespace std;
int count(const string &s)
{
    int n = 0, pos = 0;
    while ((pos = s.find(" ", pos)) != string::npos)
    {
        n++;
        pos += 1;
    }
    return ++n;
}
int main()
{
    string s; //输入的字符串
    string s1, s2;
    getline(cin, s);
    s1 = s.substr(0, s.find('-')); //左边的牌
    s2 = s.substr(s.find('-') + 1); //右边的牌
    int n1 = count(s1); //左边牌的数量
    int n2 = count(s2); //右边牌的数量
    string POKER = "345678910JQKA2";
    string MAX = "joker JOKER";
    //1.对王
    if (s1 == MAX)
        cout << s1 << endl;
    else if (s2 == MAX)
        cout << s2 << endl;
    //2.两边数量相等
    else if (n1 == n2)
    {
        string f1, f2;
        f1 = s1.substr(0, f1.find(" "));
        f2 = s2.substr(0, f2.find(" "));
        if (POKER.find(f1) > POKER.find(f2))
            cout << s1 << endl;
        else
            cout << s2 << endl;
    }
    else if (4 == n1)
        cout << s1 << endl;
    else if (4 == n2)
        cout << s2 << endl;
    else
        cout << "ERROR" << endl;
    return 0;
}
```



[点此](#)或手机扫描二维码查看代码编写过程