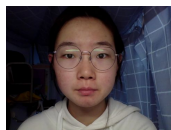


C++方向每日一题day05_4月23日-王梅-测评结果

考生信息



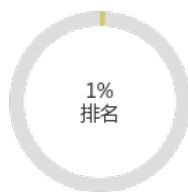
王梅

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（123.139.235.54） | 做题用时：01:15:36(2021-04-23 09:08:36开始答题，10:24:19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	45.0	9	1	00:32:57	已阅
编程	50.0	2	1	00:41:16	已阅

知识点技能图谱

其他知识点	2/2
C/C++	1/1
C++	6/7
模拟	1/1
字符串	1/1
穷举	1/1
贪心	1/1
动态规划	1/1

知识点	得分	正确题数
其他知识点	10.0	2
C/C++	5.0	1
C++	30.0	6
模拟	25.0	1
字符串	25.0	1
穷举	25.0	1
贪心	25.0	1
动态规划	25.0	1

历史笔记记录

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

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	00:30:24	C++	7ms	400K			1%
编程题2	100%	6	00:10:52	C++	23ms	760K			1%

1 [平均分4.7分 | 53人正确/56人做题 | 用时 : <1分 | 得分 : 5.0 / 5.0
以下程序的输出结果是？

```
int x = 1;
do{
    printf("%2d\n",x++);
}while(x--);
```

- A 1
- B 无任何输出
- C 2
- D 陷入死循环

他的回答： D (正确)
正确答案： D

2 [平均分2.9分 | 32人正确/55人做题 | 用时 : 3分 | 得分 : 5.0 / 5.0
下面两个结构体

```
struct One{
    double d;
    char c;
    int i;
}
struct Two{
    char c;
    double d;
    int i;
}
```

在#pragma pack(4)和#pragma pack(8)的情况下，结构体的大小分别是
A 16 24,16 24
B 16 20,16 20
C 16 16,16 24

D 16 16,24 24

他的回答： C (正确)

正确答案： C

3 [平均分4.2分 | 46人正确/55人做题 | 用时：3分  得分：5.0 / 5.0

下列程序的打印结果是？

```
char p1[15]= "abcd",*p2= "ABCD", str[50]= "xyz";
strcpy(str+2, strcat(p1+2,p2+1));
printf("%s",str);
```

- A xyabcAB
- B abcABz
- C ABabcz
- D xycdBCD
- E 运行出错

他的回答： D (正确)

正确答案： D

4 [平均分3.5分 | 38人正确/54人做题 | 用时：2分  得分：5.0 / 5.0

下面程序的输出结果是

```
#include<iostream.h>
void main(){
    int n[][3] = {10,20,30,40,50,60};
    int (*p)[3];
    p=n;
    cout<<p[0][0]<<","<<*(p[0]+1)<<","<<(*p)[2]<<endl;
}
```

- A 10,30,50
- B 10,20,30
- C 20,40,60
- D 10,30,60

他的回答： B (正确)

正确答案： B

5 [平均分1.8分 | 19人正确/53人做题 | 用时：2分  得分：5.0 / 5.0

```
int main(){
    int a;float b,c;
    scanf("%2d%3f%4f",&a,&b,&c);
    printf("\na=%d,b=%d,c=%f\n",a,b,c);
}
```

若运行时从键盘上输入9876543210I,则上面程序在gcc编译器下的输出结果是

- A a=98,b=765,c=4321.000000
- B a=98,b=0,c=0.000000
- C a=98,b=765.000000,c=4321.000000

D a=98,b=765.0,c=4321.0

他的回答： B (正确)

正确答案： B

6 [平均分2.9分 | 31人正确/54人做题 | 用时：<1分 | 得分：5.0 / 5.0

STL中的unordered_map和priority_queue使用的底层数据结构分别是什么?()

A rbtree,queue

B hashtable,heap

C rbtree,heap

D hashtable,queue

他的回答： B (正确)

正确答案： B

7 [平均分1.5分 | 16人正确/54人做题 | 用时：2分 | 得分：0.0 / 5.0

下面说法正确的是()

A 一个空类默认一定生成构造函数,拷贝构造函数,赋值操作符,引用操作符,析构函数

B 可以有多个析构函数

C 析构函数可以为virtual,可以被重载

D 类的构造函数如果都不是public访问属性,则类的实例无法创建

他的回答： C (错误)

正确答案： A

8 [平均分2.8分 | 31人正确/56人做题 | 用时：6分 | 得分：5.0 / 5.0

```
ClassA *pclassa=new ClassA[5];
delete pclassa;
```

c++语言中,类ClassA的构造函数和析构函数的执行次数分别为()

A 5,1

B 1,1

C 5,5

D 1,5

他的回答： A (正确)

正确答案： A

9 [平均分1.5分 | 16人正确/53人做题 | 用时：3分 | 得分：5.0 / 5.0

关于重载和多态正确的是

A 如果父类和子类都有相同的方法,参数个数不同,将子类对象赋给父类后,由于子类继承于父类,所以使用父类指针调用父类方法时,实际调用的是子类的方法

B 选项全部都不正确

C 重载和多态在C++面向对象编程中经常用到的方法,都只在实现子类的方法时才会使用

D

```
class A{
    void test(float a){cout<<"1";}
};
class B:public A{
    void test(int b){cout<<"2";}
};
A *a=new A;
```

```
B *b=new B;
a=b;
a.test(1.1);
结果是1
```

他的回答： B (正确)

正确答案： B

10 [平均分1.0分 | 10人正确/51人做题 | 用时：4分] 得分：5.0 / 5.0

请选择下列程序的运行结果

```
#include<iostream>
using namespace std;
class B0//基类B0声明
{
public://外部接口
virtual void display()//虚成员函数
{
    cout<<"B0::display0"<<endl;}
};
class B1:public B0//公有派生
{
public:
    void display() { cout<<"B1::display0"<<endl; }
};
class D1: public B1//公有派生
{
public:
    void display(){ cout<<"D1::display0"<<endl; }
};
void fun(B0 ptr)//普通函数
{
    ptr.display();
}
int main()//主函数
{
    B0 b0;//声明基类对象和指针
    B1 b1;//声明派生类对象
    D1 d1;//声明派生类对象
    fun(b0);//调用基类B0函数成员
    fun(b1);//调用派生类B1函数成员
    fun(d1);//调用派生类D1函数成员
}
```

- A B0::display0 B0::display0 B0::display0
- B B0::display0 B0::display0 D1::display0
- C B0::display0 B1::display0 D1::display0
- D B0::display0 B1::display0 B1::display0

他的回答： A (正确)

正确答案： A

11 ACM编程题 语言限制 [平均分18.7分 | 34人正确/53人做题 | 提交：7次] 得分：25.0 / 25.0

标题：统计回文 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【统计回文】“回文串”是一个正读和反读都一样的字符串，比如“level”或者“noon”等等就是回文串。花花非常喜欢这种拥有对称美的回文串，生日的时候她得到两个礼物分别是字符串A和字符串B。现在她非常好奇有没有办法将字符串B插入字符串A使产生的字符串是一个回文串。你接受花花的请求，帮助她寻找有多少

种插入办法可以使新串是一个回文串。如果字符串B插入的位置不同就考虑为不一样的办法。

例如：

A = “aba” ， B = “b” 。这里有4种把B插入A的办法：

- * 在A的第一个字母之前: "baba" 不是回文
- * 在第一个字母 ‘a’ 之后: "abba" 是回文
- * 在字母 ‘b’ 之后: "abba" 是回文
- * 在第二个字母'a'之后 "abab" 不是回文

所以满足条件的答案为2

输入描述：

每组输入数据共两行。
第一行为字符串A
第二行为字符串B
字符串长度均小于100且只包含小写字母

输出描述：

输出一个数字，表示把字符串B插入字符串A之后构成一个回文串的方法数

示例1：

输入
aba b
输出
2

代码片段

功能实现

	TA的	平均
总通过率	100%	74%
基本测试用例通过率	6/6 (100%)	75%
边缘测试用例通过率	4/4 (100%)	74%

代码提交统计

	TA的	平均
使用语言	C++	
做题用时	00:30:24	00:37:56
提交次数	7	4

代码执行统计

答案错误	: 1
编译错误	: 4
答案正确	: 2

代码效率

	TA的	参考
运行时间	7ms	1s
占用内存	400K	32768K

代码规范及可读性

代码规范得分	4.8
Line 9: If an else has a brace on one side, it should have it on both [readability/braces] [5]	

他的代码：

做题用时: 30 分钟 语言: C++ 运行时间: 7ms 占用内存: 400K 程序状态: 答案正确

```
#include<iostream>
using namespace std;
#include<string>
bool test(string str1) {
    int n = str1.size();
    if (n == 1) {
        return true;
    }
    else {
        for (int i = 0, j = n - 1; i < j; i++, j--) {
            if (str1[i] != str1[j]) {
```

```

return false;
}
}
return true;
}
}

int main() {
int count = 0;
string str1, str2, tmp;
cin >> str1 >> str2;
for (int i = 0; i < str1.size() + 1; i++) {
tmp = str1.substr(0, i) + str2 + str1.substr(i);
if (test(tmp)) {
count++;
}
}
cout << count << endl;
return 0;
}

#include<iostream>
#include<string>
using namespace std;
//判断是否是回文
bool IsCircletext(const string& s)
{
size_t begin = 0;
size_t end = s.size()-1;
while(begin < end)
{
if(s[begin] != s[end])
return false;
++begin;
--end;
}
return true;
}

int main()
{
string str1, str2;
getline(cin, str1);
getline(cin, str2);
size_t count = 0;
for(size_t i = 0; i <= str1.size(); i++)
{
//将字符串2插入到字符串1 的每个位置，再判断是否为回文
string str = str1;
str.insert(i, str2);
if(IsCircletext(str))
++count;
}
cout << count << endl;
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20.2分 | 37人正确/48人做题 | 提交: 6 次] 得分: 25.0 / 25.0

标题: 连续最大和 | 时间限制: 1秒 | 内存限制: 32768K | 语言限制: 不限

【连续最大和】一个数组有 N 个元素，求连续子数组的最大和。例如：[-1,2,1]，和最大的连续子数组为[2,1]，其和为 3

输入描述：

输入为两行。第一行一个整数n(1 ≤ n ≤ 100000)，表示一共有n个元素 第二行为n个数，即每个元素,每个整数都在32位int范围内。以空格分隔。

输出描述：

所有连续子数组中和最大的值。

示例1：

输入

3 -1 2 1

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

返回非零 : 1

总通过率	100%	80%	使用语言	C++	运行错误	: 1
基本测试用例通过率	6/6 (100%)	83%	做题用时	00:10:52 00:34:48	编译错误	: 3
边缘测试用例通过率	4/4 (100%)	77%	提交次数	6 5	答案正确	: 1
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	23ms	1s				
占用内存	760K	32768K				

他的代码：

做题用时: 10 分钟 语言: C++ 运行时间: 23ms 占用内存: 760K 程序状态: 答案正确

```
#include<iostream>
using namespace std;
#include<vector>
int main()
{
    int n;
    cin >> n;
    vector<int> array(n);
    for(int i = 0; i < n; i++)
    {
        cin >> array[i];
    }
    int cursum = array[0];
    int maxsum = array[0];
    for(int i = 1; i < array.size(); i++)
    {
        cursum += array[i];
        if(cursum < array[i])
            cursum = array[i];
        if(cursum > maxsum)
            maxsum = cursum;
    }
    cout << maxsum << endl;
    return 0;
}
```

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

//得到两数的较大一个
int GetMax(int a, int b)
{
    return a > b ? a : b;
}
int main()
{
    int size;
    cin >> size;
    vector<int> nums(size);
    for(size_t i = 0; i < size; i++)
    {
        cin >> nums[i];
    }
    int sum = nums[0]; //临时最大值
    int max = nums[0]; //比较之后的最大值
    for(int i = 0; i < size; i++)
    {
        sum = GetMax(sum + nums[i], nums[i]);
        if(sum >= max)
            max = sum;
    }
    cout << max << endl;
    return 0;
}
```



[点此](#)或手机扫描二维码查看代码编写过程