

C++方向每日一题day07_4月26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（123.139.168.41） |

做题用时：01:55:21(2021-04-25 22:54:32开始答题，2021-04-26 18:10:43交卷) | 作答设备：PC |

已同意诚信声明和隐私协议

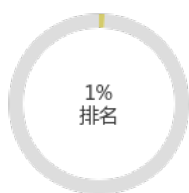
考生成绩



95.0
得分



01:55:21
用时



1%
排名

题型	得分	正确题数	排名	用时	是否阅卷
单选	45.0	9	1	00:31:14	已阅
编程	50.0	2	1	01:21:52	已阅

知识点技能图谱

其他知识点	2/2
C/C++	3/3
编程基础	1/1
C++	4/5
模拟	1/1
字符串	1/1
数学	1/1

知识点	得分	正确题数
其他知识点	10.0	2
C/C++	15.0	3
编程基础	25.0	1
C++	20.0	4
模拟	25.0	1
字符串	25.0	1
数学	25.0	1

历史笔记录

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

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%	16	00:54:12	C++	3ms	436K			1%
编程题2	100%	10	00:27:40	C++	2ms	576K			1%

1 [平均分4.2分 | 50人正确/60人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

在int p[][4]={1},{3,2},{4,5,6},{0}};中，p[1][2]的值是()

A 1
B 0
C 6
D 2

他的回答 : B (正确)
正确答案 : B

2 [平均分3.3分 | 40人正确/61人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

下列关于对象数组的描述中，()是错误的.

A 对象数组只能赋初值而不能再赋值
B 对象数组的每个元素是同一个类的对象
C 对象数组的数组名是一个指针常量
D 对象数组的下标是从0开始的

他的回答 : A (正确)
正确答案 : A

3 [平均分2.4分 | 30人正确/63人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

设变量已正确定义，以下不能统计出一行中输入字符个数（不包含回车符）的程序段是

A n=0;while(ch=getchar())!='\n')n++;
B n=0;while(getchar())!='\n')n++;
C for(n=0;getchar()!='\n';n++);

D n=0;for(ch=getchar();ch!='\n';n++);

他的回答：D (正确)

正确答案：D

4 [平均分2.2分 | 27人正确/62人做题 | 用时：2分 | 得分：5.0 / 5.0]

下列程序段执行后，输出d的值为()

```
void main()
{
    int a=1,b=0,c=-1,d=0;
    d=++a||++b&&++c;
    cout<<d<<endl;
    return;
}
```

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

他的回答：A (正确)

正确答案：A

5 [平均分2.9分 | 35人正确/61人做题 | 用时：6分 | 得分：5.0 / 5.0]

```
#include <stdio.h>
int main()
{
    int a=1,b=2,c=3,d=0;
    if(a == 1 && b++==2)
    if(b!=2||c--!=3)
        printf("%d,%d,%d\n",a,b,c);
    else
        printf("%d,%d,%d\n",a,b,c);
    else
        printf("%d,%d,%d\n",a,b,c);
    return 0;
}
```

下面程序输出是什么？

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

他的回答：D (正确)

正确答案：D

6 [平均分4.4分 | 54人正确/62人做题 | 用时：<1分 | 得分：5.0 / 5.0]

下面关于类定义的说法中，正确的是：

- A 类定义中包括数据成员和函数成员的声明
- B 类成员的缺省访问权限是保护的
- C 数据成员必须被声明为私有的
- D 成员函数只能在类体外进行定义

他的回答： A (正确)

正确答案： A

7 [平均分2.0分 | 25人正确/61人做题 | 用时：4分 | 得分：0.0 / 5.0

有如下类模板定义：()

```
template<class T> class BigNumber{
    long n;
public:
    BigNumber(T i):n(i){}
    BigNumber operator+(BigNumber b)
    {
        return BigNumber(n+b.n);
    }
};
```

已知b1,b2是BigNumber的两个对象,则下列表达式中错误的是?

- A 3+3
- B b1+3
- C b1+b2
- D 3+b1

他的回答： C (错误)

正确答案： D

8 [平均分3.7分 | 45人正确/61人做题 | 用时：2分 | 得分：5.0 / 5.0

假定一个类的构造函数为A(int aa,int bb){a=aa--;b=a*bb;}，则执行A x(4,5);语句后，x.a和x.b的值分别为()

- A 20和5
- B 3和15
- C 5和4
- D 4和20

他的回答： D (正确)

正确答案： D

9 [平均分3.9分 | 47人正确/61人做题 | 用时：4分 | 得分：5.0 / 5.0

```
#include<iostream>
#include<string>
using namespace std;
class A
{
    friend long fun(A s)
    {
        if (s.x<3) {
            return 1;
        }
        return s.x+fun(A(s.x - 1));
    }
public:
    A(long a)
    {
        x = a--;
    }
private:
```

```

    long x;
};
int main()
{
    int sum=0;
    for( int i=0; i<5; i++) {
        sum += fun(A(i));
    }
    cout<<sum;
}

```

代码运行结果是？

- A 21
- B 15
- C 9
- D 36

他的回答： B (正确)

正确答案： B

10 [平均分3.2分 | 39人正确/61人做题 | 用时：5分  得分：5.0 / 5.0

```

#include <iostream>
using namespace std;
class parent
{
    int i;
protected:
    int x;
public:
    parent(){x=0;i=0;}
    void change(){x++;i++;}
    void display();
};
class son:public parent
{
public:
    void modify();
};
void parent::display(){cout<<"x="<<x<<endl;}
void son:: modify() {x++;}
int main()
{
    son A; parent B;
    A.display();
    A.change();
    A.modify();
    A.display();
    B.change();
    B.display();
}

```

下面程序的输出是什么？

- A x=1 x=0 x=2
- B x=2 x=0 x=1
- C x=0 x=2 x=1
- D x=0 x=1 x=2

他的回答： C (正确)

正确答案：C

11 ACM编程题 语言限制 [平均分20.7分 | 43人正确/52人做题 | 提交: 16 次] 得分：25.0 / 25.0

标题：Fibonacci数列 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【Fibonacci数列】Fibonacci数列是这样定义的：

$F[0] = 0$

$F[1] = 1$

for each $i \geq 2$: $F[i] = F[i-1] + F[i-2]$

因此，Fibonacci数列就形如：0, 1, 1, 2, 3, 5, 8, 13, ...，在Fibonacci数列中的数我们称为Fibonacci数。给你一个N，你想让其变为一个Fibonacci数，每一步你可以把当前数字X变为X-1或者X+1，现在给你一个数N求最少需要多少步可以变为Fibonacci数。

输入描述：

输入为一个正整数N($1 \leq N \leq 1,000,000$)

输出描述：

输出一个最小的步数变为Fibonacci数"

示例1：

输入

15

输出

2

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 82%	使用语言	TA的 C++	平均	答案错误	8
基本测试用例通过率	6/6 (100%)	82%	做题用时	00:54:12	00:42:40	运行超时	4
边缘测试用例通过率	4/4 (100%)	82%	提交次数	16	4	编译错误	2
						答案正确	2
代码效率					代码规范及可读性		
运行时间	TA的 3ms	参考 1s	代码规范得分			4.82759	
占用内存	436K	32768K	Line 42: 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: 54 分钟 语言：C++ 运行时间：3ms 占用内存：436K 程序状态：答案正确

```
#include<iostream>
using namespace std;
int Fibonacci(int n)
{
    if(n <= 0)
    {
        return 0;
    }
    if(n == 1 || n == 2)
    {
        return 1;
```

```

}
//return Fibonacci(n - 2) + Fibonacci(n - 1);
int fn1 = 1;
int fn2 = 1;
int ret = 0;
for(int i = 3; i <= n; i++)
{
    ret = fn2 + fn1;
    fn1 = fn2;
    fn2 = ret;
}
return ret;
}
int main()
{
    int N;
    int ret;
    cin >> N;
    int min = N;
    if(N <= 21)
    {
        for (int i = 0; i < N + 6; i++)
        {
            ret = abs(Fibonacci(i) - N);
            if (min > ret)
            {
                min = ret;
            }
        }
    }
    else{
        for (int i = 9; i < N/2; i++)
        {
            ret = abs(Fibonacci(i) - N);
            if (min > ret)
            {
                min = ret;
            }
        }
        else
        {
            break;
        }
    }
    cout << min;
    return 0;
}

```

```

#include<iostream>
using namespace std;
int main()
{
    int N, f, l = 0, r = 0, f0 = 0, f1 = 1;
    cin >> N;
    while(1)
    {
        f = f0 + f1;
        f0 = f1;
        f1 = f;
        //找到比N小且距离N最近的数，求出距离
        if(f < N)
        {
            l = N - f;
        }
        else
        {
            //找到比N大且距离N最近的数，求出距离
            r = f - N;
            break;
        }
    }
    //取出最小距离
    cout << min(l, r) << endl;
    return 0;
}

```



点此或手机扫描二维码查看代码编写过程

12 完善核心代码 语言限制 [平均分19.7分 | 41人正确/52人做题 | 提交: 10 次] 得分 : 25.0 / 25.0

标题：合法括号序列判断 | 时间限制：3秒 | 内存限制：32768K | 语言限制：[Python, C++, C#, Java]

【合法括号序列判断】
对于一个字符串，请设计一个算法，判断其是否为一个合法的括号串。
给定一个字符串A和它的长度n，请返回一个bool值代表它是否为一个合法的括号串。
一个合法的括号串定义为：1.只包括括号字符；2.左括号和右括号——对应
测试样例：

"(00)",6
返回：true

测试样例：
"0a00",7
返回：false

测试样例：
"0(00",7
返回：false

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		
总通过率	TA的	平均		使用语言	TA的	平均		答案错误 : 6		
	100%	78%			C++			编译错误 : 1		
	1/1	78%			做题用时	00:27:40	00:23:36	答案正确 : 3		
	(100%)				提交次数	10	4			
代码效率						代码规范及可读性				
	TA的	参考		代码规范得分						4.6
运行时间	2ms	3s		Line 27: If/else bodies with multiple statements require braces [readability/braces] [4]						
占用内存	576K	32768K		Line 3: Add #include for string [build/include_what_you_use] [4]						

他的代码：


```
class Parenthesis {
public:
    bool chkParenthesis(string A, int n) {
        if (n % 2 == 1)
        {
            return false;
        }
        int count = 0;
        for (int i = 0; i < n; i++)
        {
            if (A[i] == '(')
            {
                count++;
            }
            else if (A[i] == ')')
            {
                if(count <= 0)
                    return false;
                else
                    count--;
            }
            else
            {
                return false;
            }
        }
        if(count == 0)
            return true;
        return false;
    }
};
```

```
class Parenthesis {
public:
    bool chkParenthesis(string A, int n) {
        stack<char> sc;
        for(auto e : A)
        {
            switch(e)
            {
                case '(' :
                    sc.push(e);
                    break;
                case ')' :
                    {
                        if(sc.empty() || sc.top() != '(')
                            return false;
                        else
                            sc.pop();
                    }
                    break;
                default:
                    return false;
            }
        }
        return true;
    };
};
```

