# 每日一题C++方向day15_11月22日测评结果

## 考生信息

王婧

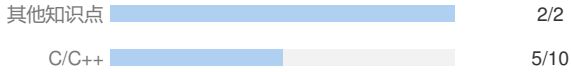
考号：2378 | 学校：财经大学 | 邮箱：1031160332@qq.com | 职位：54 |

参考区域: 陕西省西安市（113.139.120.150） | 做题用时：00:43:24(2019-11-22 12:12:57 - 12:56:30) | 作答设备： PC

## 考生成绩


| 题型 | 得分 | 正确题数 | 排名 | 用时 | 是否阅卷 |
|---|---|---|---|---|---|
| 单选 | 25.0 | 5 | 57 | 00:08:20 | -- |
| 编程 | 50.0 | 2 | 1 | 00:34:04 | -- |

## 知识点技能图谱



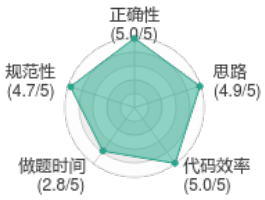

| 知识点 | 得分 | 正确题数 |
|---|---|---|
| 其他知识点 | 50.0 | 2 |
| C/C++ | 25.0 | 5 |

## 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 | 54班C/C++考试题 | 11.0% | 21.0/60 | 单选:6.0分 编程:15.0分 | 否 | 2019-09-21 17:24:48 | 2019-09-22 15:38:37 |
| 2 | 54班CPP_DS_2_考试卷 | 38.0% | 19.0/60 | 单选:4.0分 编程:15.0分 | 否 | 2019-10-31 17:40:31 | 2019-11-02 15:40:22 |
| 3 | 每日一题C++方向day02_11月7日 | 26.0% | 70.0/100 | 单选:20.0分 编程:50.0分 | 否 | 2019-11-06 16:54:27 | 2019-11-07 13:25:19 |
| 4 | 每日一题C++方向day03_11月8日 | 16.0% | 85.0/100 | 单选:35.0分 编程:50.0分 | 否 | 2019-11-07 14:40:49 | 2019-11-08 12:51:53 |
| 5 | 每日一题C++方向day04_11月9日 | 1.0% | 95.0/100 | 单选:45.0分 编程:50.0分 | 否 | 2019-11-08 11:17:08 | 2019-11-10 17:00:19 |
| 6 | 每日一题C++方向day05_11月10日 | 64.0% | 52.5/100 | 单选:25.0分 编程:27.5分 | 是，相似代码 | 2019-11-09 15:35:20 | 2019-11-10 17:49:42 |
| 7 | 每日一题C++方向day06_11月11日 | 54.000004% | 65.0/100 | 单选:40.0分 编程:25.0分 | 否 | 2019-11-09 15:38:30 | 2019-11-11 18:14:20 |
| 8 | 每日一题C++方向day07_11月13日 | 50.0% | 75.0/100 | 单选:25.0分 编程:50.0分 | 是，相似代码 | 2019-11-12 11:39:41 | 2019-11-13 14:37:22 |
| 9 | 每日一题C++方向day08_11月14日 | 8.0% | 90.0/100 | 单选:40.0分 编程:50.0分 | 是，相似代码 | 2019-11-13 10:51:46 | 2019-11-14 17:35:22 |
| 10 | 每日一题C++方向day09_11月15日 | 79.0% | 45.0/100 | 单选:20.0分 编程:25.0分 | 否 | 2019-11-14 18:14:54 | 2019-11-15 17:52:51 |
| 11 | 每日一题C++方向day10_11月16日 | 6.0% | 85.0/100 | 单选:35.0分 编程:50.0分 | 否 | 2019-11-15 13:29:43 | 2019-11-16 12:12:48 |

| 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 |
|---|---|---|---|---|---|---|---|
| 12 | 每日一题C++方向day11_11月17日 | 22.0% | 75.0/100 | 单选:25.0分 编程:50.0分 | 否 | 2019-11-16 10:59:12 | 2019-11-17 21:38:02 |
| 13 | 每日一题C++方向day12_11月18日 | 45.0% | 70.0/100 | 单选:20.0分 编程:50.0分 | 是，相似代码 | 2019-11-17 11:32:33 | 2019-11-18 17:49:04 |
| 14 | 每日一题C++方向day13_11月20日 | 2.0% | 90.0/100 | 单选:40.0分 编程:50.0分 | 是，相似代码 | 2019-11-19 11:13:30 | 2019-11-20 15:08:03 |
| 15 | 每日一题C++方向day14_11月21日 | 16.0% | 75.0/100 | 单选:25.0分 编程:50.0分 | 是，相似代码 | 2019-11-20 13:55:05 | 2019-11-21 13:15:24 |

## 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% | 8 | 00:20:00 | C++ | 4ms | 380K | 优 | 优 | 1% |
| 编程题2 | 100% | 2 | 00:14:04 | C++ | 4ms | 504K | 良 | 良 | 1% |

1 [平均分4.1分 | 117人正确/142人做题 | 用时：<1分 得分：5.0 / 5.0

对两个字符a和b进行初始化:char a[]="ABCDEF";char b[]={'A','B','C','D','E','F'};则以下叙述正确的是：

A a数组比b数组长度长
B a与b长度相同
C a与b数组完全相同
D a和b中都存放字符串

他的回答： A (正确)

正确答案： A

2 [平均分3.1分 | 88人正确/141人做题 | 用时：<1分 得分：5.0 / 5.0

x是一个行列数均为1000二维数组，下面代码效率执行最高的是（ ）

A for(int j=0;j<1000;j++) for(int i=0;i<1000;i++) x[i][j]+=x[j][i];
B for(int i=0;i<1000;j++) for(int j=0;j<1000;j++) x[i][j]+=x[j][i];
C for(int i=0;i<1000;j++) for(int j=0;j<1000;j++) x[j][i]+=x[j][i];
D for(int i=0;i<1000;i++) for(int j=0;j<1000;j++) x[i][j]+=x[i][j];

他的回答： D (正确)

正确答案： D

3 [平均分2.3分 | 66人正确/142人做题 | 用时：<1分 得分：5.0 / 5.0

C++中关于堆和栈的说法，哪个是错误的:

A 堆的大小仅受操作系统的限制，栈的大小一般一般较小
B 在堆上频繁的调用new/delete容易产生内存碎片，栈没有这个问题
C 堆和栈都可以静态分配
D 堆和栈都可以动态分配

他的回答： C (正确)

正确答案： C

4 [平均分2.1分 | 59人正确/140人做题 | 用时：<1分 得分：0.0 / 5.0

下面程序会输出什么：

```
static int a=1;
void fun1(void){    a=2;  }
void fun2(void){    int a=3;  }
void fun3(void){   static int a=4;  }
int main(int argc,char** args){
    printf(“%d”,a);
    fun1( );
    printf(“%d”,a);
    fun2( );
    printf(“%d” , a);
    fun3( );
     printf(“%d”,a);
}
```

A 1 2 3 4
B 1 2 2 2
C 1 2 2 4
D 1 1 1 4

他的回答： D (错误)

正确答案： B

---

5 [平均分2.5分 | 64人正确/130人做题 | 用时：2分 得分：5.0 / 5.0

In the main() function, after ModifyString(text) is called, what’s the value of ‘text’?

```
int FindSubString( char* pch )
{
   int   count  = 0;
   char  * p1   = pch;
   while ( *p1 != '\0' )
   {
      if ( *p1 == p1[1] - 1 )
      {
         p1++;
         count++;
      }else  {
         break;
      }
   }
   int count2 = count;
   while ( *p1 != '\0' )
   {
      if ( *p1 == p1[1] + 1 )
      {
         p1++;
         count2--;
      }else  {
         break;
      }
   }
   if ( count2 == 0 )
      return(count);
   return(0);
}
void ModifyString( char* pText )
{
   char  * p1   = pText;
   char  * p2   = p1;
   while ( *p1 != '\0' )
```

```
    {
        int count = FindSubString( p1 );
        if ( count > 0 )
        {
            *p2++ = *p1;
            sprintf( p2, "%i", count );
            while ( *p2 != '\0' )
            {
                p2++;
            }
            p1 += count + count + 1;
        }else  {
            *p2++ = *p1++;
        }
    }
}
void main( void )
{
    char text[32] = "XYBCDCBABABA";
    ModifyString( text );
    printf( text );
}
```

A XYBCDCBABABA
B XYBCBCDA1BAA
C XYBCDCBA1BAA
D XYBCDDBA1BAB

他的回答： C (正确)

正确答案： C

---

6 [平均分4.6分 | 130人正确/141人做题 | 用时：<1分 得分：5.0 / 5.0

所谓数据封装就是将一组数据和与这组数据有关操作组装在一起，形成一个集合，这集合也就是（ ）

A 类
B 对象
C 函数体
D 数据块

他的回答： A (正确)

正确答案： A

---

7 [平均分1.1分 | 31人正确/141人做题 | 用时：<1分 得分：0.0 / 5.0

关于以下代码，哪个说法是正确的？

```
myClass::foo(){
    delete this;
}
..
void func(){
    myClass *a = new myClass();
    a->foo();
}
```

A 它会引起栈溢出
B 都不正确
C 它不能编译
D 它会引起段错误

他的回答： C (错误)

正确答案： B

8 [平均分1.0分 | 28人正确/141人做题 | 用时：<1分 得分：0.0 / 5.0

假定CSomething是一个类，执行下面这些语句之后，内存里创建了____个CSomething对象。

```
CSomething a();
CSomething b(2);
CSomething c[3];
CSomething &ra = b;
CSomething d=b;
CSomething *pA = c;
CSomething *p = new CSomething(4);
```

A 10

B 9

C 8

D 7

E 6

F 5

他的回答： D (错误)

正确答案： E

9 [平均分3.0分 | 84人正确/139人做题 | 用时：<1分 得分：0.0 / 5.0

下面这段代码运行时会出现什么问题？

```
class A
{
public:
    void f()
    {
        printf("A\n");
    }
};

class B: public A
{
public:
    virtual void f()
    {
        printf("B\n");
    }
};

int main()
{
    A *a = new B;
    a->f();
    delete a;
    return 0;
}
```

A 没有问题，输出B
B 不符合预期的输出A
C 程序不正确
D 以上答案都不正确

他的回答： A (错误)

正确答案： B

---

10 [平均分1.8分 | 51人正确/139人做题 | 用时：<1分 得分：0.0 / 5.0

下面这段代码会打印出什么？

```
class A
{
public:
    A()
    {
        printf("A ");
    }

    ~A()
    {
        printf("deA ");
    }
};

class B
{
public:
    B()
    {
        printf("B ");
    }
    ~B()
    {
        printf("deB ");
    }
};

class C: public A, public B
{
public:
    C()
    {
        printf("C ");
    }
    ~C()
    {
        printf("deC ");
    }
};
int main()
{
    A *a = new C();
    delete a;
    return 0;
}
```

A A B C deA

B C A B deA
C A B C deC
D C A B deC

他的回答： D (错误)

正确答案： A

11 [平均分23.1分 | 120人正确/130人做题 | 提交: 8 次 得分：25.0 / 25.0

标题：查找输入整数二进制中1的个数 | 时间限制：1秒 | 内存限制：32768K | 语言限制： 不限

【查找输入整数二进制中1的个数】

请实现如下接口

```
public static int findNumberOf1( int num)
{
    /* 请实现 */
    return 0;
} 譬如：输入5 ，5的二进制为101，输出2
```

**涉及知识点：**

输入描述：

> 输入一个整数

输出描述：

> 计算整数二进制中1的个数

示例1：

输入

> 5

输出

> 2

**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 ：5 |
| 总通过率 | 100% | 92% | 使用语言 | C++ | | 编译错误 ：2 |
| 基本测试用例通过率 | 6/6 (100%) | 92% | 做题用时 | 00:20:00 | 00:13:29 | 答案正确 ：1 |
| 边缘测试用例通过率 | 4/4 (100%) | 92% | 提交次数 | 8 | 5 | |

| 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 | 4ms | 1s | |
| 占用内存 | 380K | 32768K | |

他的代码：

做题用时: 20 分钟 语言：C++ 运行时间：4ms 占用内存：380K 程序状态：答案正确

```
#include <iostream>
#include <bitset>
using namespace std;
int main(){
    int n;
    while(cin >> n){
```

```
        bitset<32> bs(n);
        cout << bs.count() << endl;
    }
    return 0;
}
```

12 [平均分19.9分 | 75人正确/94人做题 | 提交: 2 次 得分：25.0 / 25.0

标题：手套 | 时间限制：3秒 | 内存限制：32768K | 语言限制： [Python, C++, C#, Java]

【手套】

在地下室里放着n种颜色的手套，手套分左右手，但是每种颜色的左右手手套个数不一定相同。A先生现在要出门，所以他要去地下室选手套。但是昏暗的灯光让他无法分辨手套的颜色，只能分辨出左右手。所以他会多拿一些手套，然后选出一双颜色相同的左右手手套。现在的问题是，他至少要拿多少只手套(左手加右手)，才能保证一定能选出一双颜色相同的手套。

给定颜色种数n(1≤n≤13),同时给定两个长度为n的数组left,right,分别代表每种颜色左右手手套的数量。数据保证左右的手套总数均不超过26，且一定存在至少一种合法方案。

测试样例：

```
4,[0,7,1,6],[1,5,0,6]
```

```
返回：10(解释：可以左手手套取2只，右手手套取8只)
```

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 | 编译错误 ：1 |
| 总通过率 | 100% | 79% | 使用语言 | C++ | | 答案正确 ：1 |
| 基本测试用例通过率 | 1/1 (100%) | 79% | 做题用时 | 00:14:04 | 00:46:21 | |
| | | | 提交次数 | 2 | 3 | |

| 代码效率 | | | 代码规范及可读性 |
|---|---|---|---|
| | TA的 | 参考 | 代码规范得分 4.4 |
| 运行时间 | 4ms | 3s | Line 12: If an else has a brace on one side, it should have it on both [readability/braces] [5] |
| 占用内存 | 504K | 32768K | Line 19: Add #include for min [build/include_what_you_use] [4] |
| | | | Line 3: Add #include for vector<> [build/include_what_you_use] [4] |

他的代码：

做题用时: 14 分钟 语言：C++ 运行时间：4ms 占用内存：504K 程序状态：答案正确

```
class Gloves {
public:
    int findMinimum(int n, vector<int> left, vector<int> right) {
        // write code here
        int sum=0;
        int leftSum=0,rightSum=0;
        int leftMin=INT_MAX,rightMin=INT_MAX;
        for(int i=0;i<n;i++)
        {
            if(left[i]*right[i]==0)
                sum+=(left[i]+right[i]);
```

```
            else{
                leftSum+=left[i];
                rightSum+=right[i];
                leftMin=min(leftMin,left[i]);
                rightMin=min(rightMin,right[i]);
            }
        }
        return sum+min(leftSum-leftMin+1,rightSum-rightMin+1)+1;
    }
};
```