CCF 编程能力等级认证
Grade Examination of Software Programming

C++ 二级

2025 年 06 月

1 单选题（每题 2 分，共 30 分）

第 1 题 2025年4月19日在北京举行了一场颇为瞩目的人形机器人半程马拉松赛。比赛期间，跑动着的机器人会利用身上安装的多个传感器所反馈的数据来调整姿态、保持平衡等，那么这类传感器类似于计算机的( )。

- **A.** 处理器
- **B.** 存储器
- **C.** 输入设备
- **D.** 输出设备

第 2 题 小明购置的计算机使用一年后觉得内存不够用了，想购置一个容量更大的内存条，这时他需要的内存条是（ ）。

- **A.** RAM
- **B.** ROM
- **C.** CACHE
- **D.** EPROM

第 3 题 下面C++代码执行后的输出是（ ）。

```
int a=3;
float b = 3.5;
cout << (a *= b);
```

- **A.** 3
- **B.** 3.5
- **C.** 10
- **D.** 11

第 4 题 下面C++代码用于获得正整数的第3位数，如1234则输出2。如果是一位数或两位数，则输出0。横线处应填入的代码是( )。

```
int N, remainder;
cout << "请输入正整数:";
cin >> N;
cout << _________________;
```

- [ ] **A.** N % 1000 / 100
- [ ] **B.** N / 1000 % 100
- [ ] **C.** N / 1000 / 100
- [ ] **D.** N % 100 / 100

第 5 题 下面C++代码执行，其输出是(　　)。

```
int a, b = (6, 28);
b == a;
a = b;
cout << a << ' ' << b;
```

- [ ] **A.** 6 28
- [ ] **B.** 6 6
- [ ] **C.** 28 6
- [ ] **D.** 28 28

第 6 题 今天星期六，其后第N天星期几？如果是星期一到星期六输出形如：星期1、星期2等，星期天则输出星期天。下面的C++代码用于完成上述要求，横线处应填上的代码是(　　)。

```
int N, remainder;
cin >> N;
remainder = _____________;
if(remainder == 0)
    printf("星期六后第%d天是星期天\n", N);
else
    printf("星期六后第%d天是星期%d\n", N, remainder);
```

- [ ] **A.** `(N + 6) / 7`
- [ ] **B.** `(N + 6) // 7`
- [ ] **C.** `N % 7`
- [ ] **D.** `(N + 6) % 7`

第 7 题 下面的C++代码执行后其输出是(　)。

```
int i, Sum = 0;
for (i = 1; i < 10; i++){
    Sum += i;
    if(i % 2) continue;
    if(i % 7) break;
}
cout << Sum;
```

- [ ] **A.** 45
- [ ] **B.** 28

- [ ] **C.** 3
- [ ] **D.** 0

第 8 题 下面C++代码执行后其输出是(　　)。

```
int i, j;
for(i = 1; i < 12; i++)
    for(j = 1; j < i; j++)
        if(i * j % 2 == 1)
            break;
cout << i * j;
```

- [ ] **A.** 110
- [ ] **B.** 22
- [ ] **C.** 12
- [ ] **D.** 3

第 9 题 下面C++代码执行后输出是(　)。

```
int i, cnt = 0;
for(i = -99; i < 100; i += 2)
    cnt = 1 + cnt;
cout << cnt;
```

- [ ] **A.** 101
- [ ] **B.** 100
- [ ] **C.** 99
- [ ] **D.** 98

第 10 题 下面C++代码执行后输出是（　）。

```
int i;
for(i = 1; i < 10; i++){
    if(i % 3 != 0){
        printf("A#");
        continue;
    }
    else
        break;
    printf("0#");
}
if(i == 10) cout << "1";
```

- [ ] **A.** A#A#
- [ ] **B.** A#0#A#0
- [ ] **C.** A#A#1
- [ ] **D.** A#0#A#0#1

第 11 题 下面C++代码执行后的输出是（　）。

```
int i,j;
for(i = 0; i < 3; i++)
    for(j = 0; j < i; j++)
        printf("%d#%d-", i, j);
printf("END");
```

- [ ] **A.** 0#0-1#0-2#0-2#1-END
- [ ] **B.** 0#0-1#0-1#1-2#0-2#1-2#2-3#0-3#1-3#2-END
- [ ] **C.** 0#0-1#0-1#1-2#0-2#1-2#2-END
- [ ] **D.** 1#0-2#0-2#1-END

第 12 题 下面C++代码执行后，将输出不能被3整除且除以5余数为2的数。下列选项不能实现的是（　　）。

```
int i,j;
for(i = 0; i < 100; i++)
    if(_____________________)
        cout << i << endl;
```

- [ ] **A.** (i % 3 != 0) && (i % 5 == 2)
- [ ] **B.** (i % 3) && (i % 5 == 2)
- [ ] **C.** (i % 3) && !(i % 5 != 2)
- [ ] **D.** !(i % 3) && (i % 5 == 2)

第 13 题 下面C++代码用于判断一个大于0的正整数是几位数，横线处应填入代码先后是（　）。

```
int N, cnt;
cout << "请输入大于0的正整数: ";
cin >> N;

cnt = 0;
while  (_________){
    cnt += 1;
    _____________;
}

cout << cnt;
```

- [ ] **A.**

```
N > 1
N = N / 10
```

- [ ] **B.**

```
N > 1
N /= 10
```

- [ ] **C.**

```
N == 0
N /= 10
```

- [ ] **D.**

```
N > 0
N /= 10
```

第 14 题 判断一个数是否为自守数。自守数的定义是如果一个数的平方其尾数与该数相同，则为自守数，如25的平方是625，其尾数是25，所以25是自守数。相关说法错误的是（ ）。

```
int N, N1, M1;
cout << "输入一个正整数: ";
cin >> N;
N1 = N, M1 = N * N;

bool Flag = true;

while (N1 > 0){
    if (N1 % 10 != M1 % 10){
        Flag = false;
        break;
    }
    else{
        N1 = N1 / 10, M1 = M1 / 10;
    }
}

if (Flag == true)
    printf("%d的平方是%d,是自守数", N, N * N);
else
    printf("%d的平方是%d,不是自守数", N, N * N);
```

- [ ] **A.** 如果Flag在循环中不被改为false，则说明该数是自守数
- [ ] **B.** 代码 `if (N1 % 10 != M1 % 10)` 用于判断其个位数是否相等，如果不等，则表明不是自守数
- [ ] **C.** 代码 `N1 = N1 / 10, M1 = M1 / 10` 将个位数去掉
- [ ] **D.** 将 `N1 > 0` 改为 `N > 0` 效果相同

第 15 题 下面C++代码实现输出如下图形，相关说法错误的是（ ）。

```
请输入层数: 10
0
12
345
6789
01234
567890
1234567
89012345
678901234
5678901234
```

```
int line_number, now_number;
int i,row;

cout << "请输入行数: ";
cin >> line_number;
now_number = 0;

for (row = 1; row < line_number + 1; row++){ // L1
    for (i = 0; i < row; i++){ // L2
        cout << now_number;
        now_number += 1;
        if (now_number == 10) now_number = 0; // L3
```

```
    }
    cout << endl;
}
```

- [ ] **A.** 代码 `now_number = 0` 移动到 `L1` 和 `L2` 标记的两行代码之间，效果维持不变
- [ ] **B.** 代码 `now_number += 1` 修改为 `now_number = 1 + now_number` 效果维持不变
- [ ] **C.** 将代码 `now_number == 10` 调整为 `now_number > 9` 效果维持不变
- [ ] **D.** 将最后一行的 `cout << endl` 修改为 `cout << "\n"` ，效果维持不变

2 判断题（每题 2 分，共 20 分）

第 1 题 人们现在参加很多闭卷考试时通常都不允许带智能手机、平板电脑等，此外很多种智能手表同样因为具有嵌入操作系统及通信等功能，所以也不允许随身携带。（ ）

第 2 题 在C++代码中，假设N为正整数，则 `N / 10` 舍弃个位数。如果 `N` 小于10，则其值为0，大于10则是舍弃个位数的数。(　　)

第 3 题 下列C++代码执行后，其输出为 `10 20` ，即 `a == b` 和 `b == a` 对 `a` 和 `b` 的值没有任何影响。(　　)

```
int a = 10, b = 20;
a == b;
b == a;
cout << (a, b);
```

第 4 题 `a` 和 `b` 分别是C++的整型变量，如果表达式 `max(a, b) == min(a, b)` 的值为真，则说明 `a` 和 `b` 相等。（ ）

第 5 题 下面C++代码编译时将报错，因为字符变量 `a` 被赋值了浮点值。(　　)

```
char a = '1';
a = 45.6;
cout << a;
```

第 6 题 下面C++代码执行时如输入59.99，将输出 及格 两个汉字。（ ）

```
int score;
cout << "请输入学生成绩: ";
cin >> score;
if (score < 60)
    cout << "不及格";
else
    cout << "及格";
```

第 7 题 在下面的C++代码中，因为 `continue` 将被执行，因此不会有输出。（ ）

```
int i;
for (i = 1; i < 10; i++)
    if (i % 2 == 0)
        continue;
if(i == 10)
    cout << "END";
```

第 8 题 下面的C++代码执行后将输出15。（ ）

```
int Sum = 0;
for (int i = 0; i < 5; i++)
    Sum += i;
cout << Sum;
```

第 9 题 将下面C++代码中的 `(int i = 5; i > 1; i--)` 调整为 `(int i = 1; i < 5; i++)` 输出结果相同，因为5到1与1到5的求和相同。（ ）

```
int tnt;
tnt = 0;
for (int i = 5; i > 1; i--)
    tnt += i;
cout << tnt;
    cout << endl;
```

第 10 题 为实现如下效果，即N行N列字符。当输入是奇数时，中间列为 `*`，其他是-；当输入是偶数时，则中间两列是 `*`，其他是-。字符阵列后的代码能实现其效果。（ ）

```
/*
5
--*--
--*--
--*--
--*--
--*--
*/
```

```
int N;
cin >> N;

int i,j;
for (i = 0; i < N; i++){
    for (j = 0; j < N; j++)
        if ((j == N / 2) || (j == (N-1) / 2))
            cout << "*";
        else
            cout << "-";
    cout << endl;
}
```

3 编程题（每题 25 分，共 50 分）

3.1 编程题 1

- **试题名称**：数三角形
- **时间限制**：1.0 s
- **内存限制**：512.0 MB

3.1.1 题目描述

直角三角形有两条直角边与一条斜边，设两条直角边的长度分别为 a, b，则直角三角形的面积为 $\frac{ab}{2}$。

请你计算当直角边长 a, b 均取不超过 n 的正整数时，有多少个不同的面积为整数的直角三角形。直角边长分别为 a, b 和 a', b' 的两个直角三角形相同，当且仅当 $a = a', b = b'$ 或者 $a = b', b = a'$。

3.1.2 输入格式

一行，一个整数 n，表示直角边长的最大值。

3.1.3 输出格式

输出一行，一个整数，表示不同的直角三角形数量。

3.1.4 样例

3.1.4.1 输入样例 1

```
3
```

3.1.4.2 输出样例 1

```
3
```

3.1.4.3 输入样例 2

```
5
```

3.1.4.4 输出样例 2

```
9
```

3.1.5 数据范围

对于所有测试点，保证 $1 \leq n \leq 1000$。

3.2 编程题 2

- **试题名称**：幂和数
- **时间限制**：1.0 s
- **内存限制**：512.0 MB

3.2.1 题目描述

对于正整数 n，如果 n 可以表为两个 2 的次幂之和，即 $n = 2^x + 2^y$（x, y 均为非负整数），那么称 n 为幂和数。

给定正整数 l, r，请你求出满足 $l \leq n \leq r$ 的整数 n 中有多少个幂和数。

3.2.2 输入格式

一行，两个正整数 l, r，含义如上。

3.2.3 输出格式

输出一行，一个整数，表示 l, r 之间幂和数的数量。

3.2.4 样例

3.2.4.1 输入样例 1

```
2 8
```

3.2.4.2 输出样例 1

```
6
```

3.2.4.3 输入样例 2

```
10 100
```

3.2.4.4 输出样例 2

```
20
```

3.2.5 数据范围

对于所有测试点，保证 $1 \leq l \leq r \leq 10^4$。