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

C++ 二级

2024 年 12 月

1 单选题（每题 2 分，共 30 分）

第 1 题 2024年10月8日，诺贝尔物理学奖“意外地”颁给了两位计算机科学家约翰·霍普菲尔德（John J. Hopfield）和杰弗里·辛顿（Geoffrey E. Hinton）。这两位科学家的主要研究方向是（ ）。

- **A.** 天体物理
- **B.** 流体力学
- **C.** 人工智能
- **D.** 量子理论

第 2 题 计算机系统中存储的基本单位用B来表示，它代表的是（ ），比如某个照片大小为3MB。

- **A.** Byte
- **B.** Block
- **C.** Bulk
- **D.** Bit

第 3 题 C++语句 `cout << (3 + 3 % 3 * 2 - 1)` 执行后输出的值是（ ）。

- **A.** -1
- **B.** 4
- **C.** 56
- **D.** 2

第 4 题 下面C++代码执行后其输出是( )。

```
for (int i=0; i<10; i++)
    printf("%d",i);
```

- **A.** 123456789
- **B.** 0123456789
- **C.** 12345678910

☐ **D.** 012345678910

第 5 题 下面C++代码的相关说法中，正确的是(　　)。

```
int tnt;
for (int i=0; i<10; i++)
    tnt += i;
cout << tnt;
```

☐ **A.** 上述代码执行后其输出相当于求1-10的和（包含10）

☐ **B.** 上述代码执行后其输出相当于求1-10的和（不包含10）

☐ **C.** 上述代码执行后其输出相当于求0-10的和（不包含10）

☐ **D.** 上述代码执行后将输出不确定的值

第 6 题 下面C++代码执行后输出是（　）。

```
int i;

for (i=1; i<10; i++)
    if (i % 2)
        continue;
    else
        break;

cout << i;
```

☐ **A.** 1

☐ **B.** 2

☐ **C.** 9

☐ **D.** 10

第 7 题 下面C++代码执行后的输出是（　）。

```
for (i=0; i<10; i++){
    if (i % 3)
        continue;
    printf("0#");
}
if(i>=10)
    printf("1#");
```

☐ **A.** 0#0#0#0#0#0#0#1#

☐ **B.** 0#0#0#0#0#0#1#

☐ **C.** 0#0#0#0#1#

☐ **D.** 0#0#0#0#

第 8 题 下面C++代码用于输出0-100之前（包含100）能被7整除但不能被3整除的数，横线处不能填入的代码是（）。

```
for (i=0; i<100; i++)
    if(_____________)
        cout << i << endl;
```

- [ ] **A.** i % 7 == 0 && i % 3 != 0
- [ ] **B.** !(i % 7) && i % 3 != 0
- [ ] **C.** i % 7 && i % 3
- [ ] **D.** i % 7 == 0 && !(i % 3 == 0)

第 9 题 下面C++代码用于求正整数各位数字之和，横线处不应填入代码是（ ）。

```
int tnt, N;
printf("请输入正整数: ");
cin >> N;
tnt = 0;
while (N != 0){
    _______________
    N /= 10;
}
cout <<tnt;
```

- [ ] **A.** tnt = tnt + N % 10
- [ ] **B.** tnt += N % 10
- [ ] **C.** tnt = N % 10 + tnt
- [ ] **D.** tnt = N % 10

第 10 题 下图的C++程序执行后的输出是（ ）。

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

- [ ] **A.** 0010120123
- [ ] **B.** 01012012301234
- [ ] **C.** 001012012301234
- [ ] **D.** 01012012301234012345

第 11 题 下面C++代码用于实现图示的九九乘法表。相关说法错误的是（ ）。

```
/*
1*1=1
1*2=2  2*2=4
1*3=3  2*3=6  3*3=9
1*4=4  2*4=8  3*4=12 4*4=16
1*5=5  2*5=10 3*5=15 4*5=20 5*5=25
1*6=6  2*6=12 3*6=18 4*6=24 5*6=30 6*6=36
1*7=7  2*7=14 3*7=21 4*7=28 5*7=35 6*7=42 7*7=49
1*8=8  2*8=16 3*8=24 4*8=32 5*8=40 6*8=48 7*8=56 8*8=64
```

```
1*9=9  2*9=18 3*9=27 4*9=36 5*9=45 6*9=54 7*9=63 8*9=72 9*9=81
*/

for (int Hang=1; Hang<10; Hang++){
    for (int Lie=1; Lie<Hang+1; Lie++){
        if (Lie * Hang > 9)
            printf("%d*%d=%d ", Lie, Hang, Lie*Hang);
        else
            printf("%d*%d=%d  ", Lie, Hang, Lie*Hang);
        // L2
    }

    printf("\n"); // L1
}
```

- [ ] **A.** 将L1注释的 `printf("\n")` 移到L2注释所在行，效果相同
- [ ] **B.** 将L1注释的 `printf("\n")` 修改为 `print("%c", '\n')` 效果相同
- [ ] **C.** 将 `Lie * Hang > 9` 修改为 `Lie * Hang >= 10` 效果相同
- [ ] **D.** 将 `Lie * Hang > 9` 修改为 `Hang * Lie > 9` 效果相同

第 12 题 在数学中N!表示N的阶乘，即1到N的乘积，如 `3!=1*2*3` 。下面的C++用于求1-N的阶乘之和，如N为3，则是 `1!+2!+3!` 。下面代码段补充选项后用于实现上述功能，其中不能实现阶乘和的选项是（ ）。

```
int N;

cin >> N;

int tnt=0, nowNum = 1; //tnt保存求和之值，当前N的阶乘

for (int i=1; i < N + 1; i++){
    _____________ // 基于上一个计算出当前数的阶乘
    _____________ // 从1到i每个数阶乘之和
}

cout << tnt;
```

- [ ] **A.**

```
nowNum *= i;
tnt += nowNum;
```

- [ ] **B.**

```
nowNum = nowNum * i;
tnt = tnt + nowNum;
```

- [ ] **C.**

```
nowNum *= i;
tnt = nowNum + tnt;
```

- [ ] **D.**

```
nowNum = nowNum + i;
tnt *= nowNum;
```

第 13 题 下面C++代码用于输出N和M之间（可以包括N和M）的孪生素数。孪生素数是指间隔为2的两个数均为素数，如11和13分别是素数，且间隔为2。isPrime(N)用于判断N是否为素数的函数。为完成上述功能，横线处应填上的代码是（ ）。

```
int N,M;

//本题假设N小于M
cin >> N >> M;

for (int i = N; i < __________; i++)
    if (isPrime(i) && isPrime(i + 2))
        printf("%d %d\n",i, i + 2);
```

- ☐ **A.** M - 2
- ☐ **B.** M - 1
- ☐ **C.** M
- ☐ **D.** M + 1

第 14 题 下面C++代码实现输出如下图形，横线应填入的代码是（ ）。

```
/*
高度:5
    *
   ***
  *****
 *******
*********
*/

int height;
cout << "高度: ";

//获取用户输入的高度
cin >> height;
for (i=0; i<height; i++){
    //打印每行前面的空格
    for (j = 0; j < _________________; j++)
        cout << " ";
    //打印每行的星号
    for (k = 0; k < _________________; k++)
        cout << "*";
    //输出一行后，换行
    cout << endl;
}
```

- ☐ **A.**

```
height - i
2 * i
```

☐ **B.**

```
height
2 * i
```

☐ **C.**

```
height - i
2 * i + 1
```

☐ **D.**

```
height - i - 1
2 * i + 1
```

第 15 题 下面C++代码执行后的输出是30，则横线处不能填入（ ）。

```
int a=10,b=20,c=30;

cout << ____________________ << endl;

cout << endl;
```

☐ **A.** `max(max(a, b), c)`

☐ **B.** `min(a+b, c)`

☐ **C.** `sqrt(a+b+c)`

☐ **D.** `(a+b+c)/2`

2 判断题（每题 2 分，共 20 分）

第 1 题 在Windows的资源管理器中为已有文件A建立副本的操作是Ctrl+C，然后Ctrl+V。( )

第 2 题 在C++代码中，假设N为正整数，则 `cout << (N - N / 10 * 10)` 将获得N的个位数。( )

第 3 题 在C++语句 `cout << (10 <= N <= 12)` 中，假设N为12，则其输出为1。( )

第 4 题 如果C++表达式 `int(sqrt(N))*int(sqrt(N)) == N` 的值为True，则说明N为完全平方数，如4、9、25等。（ ）

第 5 题 下面C++代码执行后将输出2*3=6。（ ）

```
int a = 2, b = 3;
printf("%%a*%%b=%d",a*b);
```

第 6 题 以下C++代码因为循环变量为将导致错误，即不能作为变量名称，不符合C++变量命名规范。（ ）

```
for (int _ = 0; _ < 10; _++)
    continue;
```

第 7 题 下面C++代码执行后因为有break，将输出0。（ ）

```
int i;
for (i = 0; i < 10; i++){
    continue;
    break;
}
cout << i;
```

第 8 题 下面的C++代码执行后将输出18行“OK”。（ ）

```
int i,j;
for (i = 8; i > 2; i-=2)
    for (j =0; j < i; j++)
        printf("OK\n");
```

第 9 题 将下面C++代码中的 `i = 1` 调整为 `i = 0` 的输出结果相同。（ ）

```
int i;
int cnt = 0;
for (i = 1; i < 5; i++)
   if(i%2) cnt += 1;
cout << cnt;
```

第 10 题 下面两段C++代码都是用于求1-10的和，其运行结果相同。通常说来，for循环都可以用while循环实现。（ ）

```
int tnt;
int i;

tnt = 0;
for (i = 1; i < 10 + 1; i++)
    tnt += i;
cout << tnt << endl;
```

```
int tnt;
int i;

tnt = 0;
i = 1;
while (i <= 10){
    tnt += i;
    i += 1;
}
cout << tnt << endl;
```

3　编程题（每题 25 分，共 50 分）

3.1　编程题 1

- **试题名称**：寻找数字
- **时间限制**：1.0 s
- **内存限制**：512.0 MB

3.1.1　题面描述

小杨有一个正整数 a，小杨想知道是否存在一个正整数 b 满足 $a = b^4$。

3.1.2　输入格式

第一行包含一个正整数 t，代表测试数据组数。

对于每组测试数据，第一行包含一个正整数代表 a。

3.1.3　输出格式

对于每组测试数据，如果存在满足条件的正整数 b，则输出 b，否则输出 -1。

3.1.4　样例

```
3
16
81
10
```

```
2
3
-1
```

对于全部数据，保证有 $1 \leq t \leq 10^5, 1 \leq a \leq 10^8$。

3.2 编程题 2

- **试题名称**：数位和
- **时间限制**：1.0 s
- **内存限制**：512.0 MB

3.2.1 题面描述

小杨有 n 个正整数，小杨想知道这些正整数的数位和中最大值是多少。

“数位和”指的是一个数字中所有数位的和。例如：

对于数字 12345，它的各个数位分别是 1, 2, 3, 4, 5。将这些数位相加，得到：

$$1+2+3+4+5=15$$

因此，12345 的数位和是 15。

3.2.2 输入格式

第一行包含一个正整数 n，代表正整数个数。

之后 n 行，每行包含一个正整数。

3.2.3 输出格式

输出这些正整数的数位和的最大值。

3.2.4 样例

```
3
16
81
10
```

```
9
```

对于全部数据，保证有 $1 \leq n \leq 10^5$，每个正整数不超过 10^{12}。