



# 2023 NOC 软件创意编程初中组 C++ 初赛

## 一、单选

1 (2分) 下列选项中，不属于计算机软件系统的是：

- A. 操作系统
- B. 系统实用程序
- C. 用户自己开发的软件系统
- D. 存储系统

2 (2分) 十进制数17对应的二进制数是 ( )

- A. 11000
- B. 10001
- C. 11111
- D. 10101

3 (2分) 关于函数，以下说法错误的是 ( )

- A. 函数可以没有返回值
- B. 函数可以没有参数
- C. 函数名只能包含数字、字母和下划线
- D. 调用函数时参数个数对应即可，类型随意

4 (2分) 下列选项中，关于变量命名的描述正确的是：

- A. 可以设置一个变量，其变量名为int
- B. 变量在命名时可以包含数字
- C. 变量在命名时必须以字母开头
- D. 变量在命名时可以包含空格

5 (2分) 阅读以下代码，正确的输出是 ( )

```
char ch = 'd';  
ch = ch - 'a' + 'A';  
cout << ch;
```

- A. 4
- B. 3
- C. D
- D. E

6 (2分) 阅读以下程序片段，正确的输出是：

```
int a[100];
```



```
for (int i = 1; i <= 5; i++)
{
    cin >> a[i];
}
for (int i = 1; i <= 5; i++)
{
    if (i % 2 == 0)
    {
        cout << a[i / 2] << " ";
    }
}
```

- A. 若输入数据依次为 1 2 3 4 5 则输出为 1 2
- B. 若输入数据依次为 1 2 3 4 5 则输出为 2 4
- C. 若输入数据依次为 1 2 3 4 5 则输出为 1 3
- D. 若输入数据依次为 1 2 3 4 5 则输出为 2 3

7 (2分) 阅读以下程序片段，判断该函数无法完成什么功能：（其中len为数组中元素个数）

```
void fun(int arr[], int len)
{
    for (int i = 0; i < len - 1; i++)
        for (int j = 0; j < len - 1 - i; j++)
            if (arr[j] > arr[j + 1])
            {
                swap(arr[j], arr[j + 1]);
            }
}
```

- A. 将数组中元素从小到大排序
- B. 将数组中元素从大到小排序
- C. 找到数组中最大值
- D. 找到数组中最小值



8 (2分) 阅读下列语句, 下列语句在运行后, 能保证a的值为2的是:

- A. `int a = int(1.6);`    B. `int a = int('2');`    C. `int a = '2';`    D. `int a = 1.25 * 2;`

9 (2分) `int fun(int x, int y);`

不可与下列哪个函数构成重载 ()

- A. `int fun(int x, int y, int z);`    B. `double fun(int x, int y);`  
C. `double fun(double x, double y);`    D. `double fun(int x, double y);`

10 (2分) 关于一个长度为 n 的数组, 以下说法错误的是:

- A. 删除一个元素时间复杂度是 $O(n)$     B. 访问一个元素时间复杂度是 $O(n)$   
C. 插入一个元素的时间复杂度为 $O(n)$     D. 数组元素地址连续

11 (2分) 甲、乙、丙三位同学选修课程, 共4门课程, 甲选修1门, 乙丙各选修2门, 则不同的选修方案有 ( ) 种

- A. 64    B. 32    C. 144    D. 256

12 (2分) 对于入栈序列为a, b, c, d, e, f的序列, 以下不可能合法的出栈序列是 ( )

- A. a, b, c, d, e, f    B. a, b, d, c, f, e    C. c, b, a, f, e, d    D. a, d, b, c, e, f

13 (2分) 有如下程序段, 变量a、b、sum均为已赋值正整数

```
sum = a;  
for (int i = 1; i <= b; i++)  
    sum += 1;
```

下列语句与上述程序段修改sum值的功能等价的赋值语句是 ( )

- A. `sum = a + b;`    B. `sum += b;`    C. `sum += a;`    D. `sum += a * b;`

14 (2分) 以下选项中结果为true的是 ( )

```
int a = 13, b = 11, c = 0;
```



- A.  $(a == 13 \parallel b != 11) \&\& c$                       B.  $a <= b \parallel c \parallel a == b$   
C.  $a + b > c \&\& !c \&\& a - b > c$                       D.  $a != b \&\& a * c \parallel b < 0$

15 (2分) 阅读程序

```
int fac(int n)
{
    if (n == 0)
        return 1;
    else
        return n * fac(n - 1);
}
```

输出fac(4)结果是 ( )

- A. 196                      B. 24                      C. 120                      D. 12

16 (2分) 阅读以下程序片段，下列选项中是程序正确输出的是：

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     for (int i = 1; i <= 3; i++)
7     {
8         for (int j = 1; j <= i + 1; j++)
9         {
10            cout << "* ";
11        }
12        cout << endl;
13    }
14    return 0;
15 }
```



A. \*\*

\*\*\*

\*\*\*\*

C. \*

\*\*

\*\*\*

B. \* \*

\* \* \*

\* \* \* \*

D. \*

\* \*

\* \* \*

17 (2分) 下列满足FILO (先进后出) 的数据结构是 :

A. 队列

B. 栈

C. 链表

D. 优先队列

18 (2分) 判断表达式 :  $(a == 0) \parallel (a != 0) \&\& (a > 0) \parallel (a < 0)$  的值 :

A. true

B. false

C. a 为 1 答案为true , a为0 答案为false

D. 无法判断

19 (2分) 以下对多维数组的描述语句中 , 不正确的是 ( )

A. 各个维度的下标均是从0开始编号的

B. 每个元素类型必须完全相同

C. 数组所占的内存大小由数据类型和数组每一维的大小共同决定

D. 多维数组在内存中是分段存储的

20 (2分) 阅读以下程序片段 , 请问正确的输出结果是

```
int t = 10001, x = 0, y = 0;
```

```
while (t > 0)
```

```
{
```

```
    t--;
```

```
    x = t % 8;
```

```
    if (x == 1)
```



```
    y++;  
}  
cout << y << endl;
```

- A. 1249                      B. 1250                      C. 1251                      D. 1252

## 二、判断

21 (2分) 2\_sum是一个合法的C++变量名。

- A. 对    B. 错

22 (2分) C++中的函数可以同时返回多个相同或者不同类型的返回值。

- A. 对    B. 错

23 (2分) 广度优先搜索需要用到队列的数据结构，可以用来求最短路径。

- A. 对    B. 错

24 (2分) 二分搜索算法的基本思想是将  $n$  个升序排序的元素分成个数大致相同的两半，取  $a[n/2]$  与  $x$  进行比较：如果  $x < a[n/2]$ ，则只要在数组  $a$  的左半部继续搜索  $x$ 。

- A. 对    B. 错

25 (2分) 对于一棵具有  $n$  个结点，高度为  $h$  的任何二叉树，进行任一种次序遍历的时间复杂度均为  $O(h)$ 。

- A. 对    B. 错

## 三、多选

26 (4分) 现有一字符串 `string s = "Hello NOC!"`，下列关于该字符串的描述中，正确的是：



- A. `s.substr(0, 3)` 的结果是 "Hell"
- B. `s[s.size()-1]` 的结果是 'C'
- C. `s.substr(0, 5) < s.substr(6, 3)` 的结果是 true
- D. `s.substr(3, 1)` 的结果是 "l"

27 (4分) 现有两个已定义的 int 类型变量 a, b, 大小范围为 (1 <= a, b <= 100), 下列选项中, 能够实现交换a, b两变量的值的是:

- |                               |                            |                            |                            |
|-------------------------------|----------------------------|----------------------------|----------------------------|
| A. <code>a = b; b = a;</code> | B. <code>int t = a;</code> | C. <code>a = a - b;</code> | D. <code>a = a + b;</code> |
|                               | <code>a = b;</code>        | <code>b = a + b;</code>    | <code>a = a - b;</code>    |
|                               | <code>b = t;</code>        | <code>a = b - a;</code>    | <code>b = a - b;</code>    |

28 (4分) 阅读以下程序片段, 下列选项中描述错误的是:

```
int m;  
cin >> m;  
while (m % 3 == 0)  
{  
    m /= 3;  
}  
while (m % 5 == 0)  
{  
    m /= 5;  
}  
cout << m << endl;
```

- A. 若输入的数据是质数, 则输出的数与输入的数相同。
- B. 若输入的数为 15, 则输出的结果为 1
- C. 若输入的数为 100, 则输出的结果为 4
- D. 若输入的数为 225, 则输出的结果为 15



29 (4分) 对于长度为n的数组, 排序算法的最坏时间复杂度为 $O(n^2)$ 的是

- A. 选择排序      B. 归并排序      C. 插入排序      D. 冒泡排序

30 (4分) 阅读以下程序

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

```
int cnt[10] = {};
```

```
void split(int x)
```

```
{
    while (x > 0)
    {
        cnt[x % 10]++;
        x /= 10;
    }
}
```

```
int main()
```

```
{
    for (int i = 1; i <= 5; i++)
    {
        int x;
        cin >> x;
        split(x);
    }
    int ans = 0;
    for (int i = 0; i <= 9; i++)
    {
        ans = max(ans, cnt[i]);
    }
}
```





```
}  
    cout << ans << endl;  
    return 0;  
}
```

有关上述程序，说法正确的是 ( )

- A. 假设输入的5个整数都是0，则输出的结果为0
- B. 假设输入的5个整数都是0，则输出的结果为5
- C. 假设输入的5个整数是{10, 10, 20, 40, 33}，则输出的结果是4
- D. 该程序可以统计输入的5个int范围内的正整数中，数字0~9出现次数最多的次数

## 四、填空题

31 (3分) 对数组进行sort排序，reverse反转，二分查找lower\_bound、upper\_bound等操作需要包含头文件：< \_\_\_\_\_ > (本题横线处仅填写英文字母)

32 (3分) C++语言中的每条基本语句以 \_\_\_\_\_ 作为结束符。(请填英文符号)

33 (3分) 假定一个二维数组的定义为 char ch[8][8]; 则该数组所占存储空间的字节总数为 \_\_\_\_\_

34 (3分) 假设字符串 s = "NOC23"，则 s 的非空子串的个数为 \_\_\_\_\_

35 (3分) int fun(int a, int b)

```
{  
    if (b == 0) return a;  
    return fun(b, a % b);  
}
```

执行 fun(84, 119) 函数返回值为 \_\_\_\_\_



36 (3分) 已知二叉树的前序遍历顺序为ABDCEGHF，中序遍历顺序为DBAGEHCF，该二叉树的后序遍历为 \_\_\_\_\_

37 (3分) #include <iostream>

```
using namespace std;
```

```
struct point
```

```
{
```

```
    int x, y;
```

```
};
```

```
void add(point &a, point &b)
```

```
{
```

```
    a.x += b.x;
```

```
    a.y += b.y;
```

```
}
```

```
int main()
```

```
{
```

```
    point a, b;
```

```
    cin >> a.x >> a.y >> b.x >> b.y;
```

```
    add(a, b);
```

```
    add(a, a);
```

```
    cout << a.x << ' ' << a.y << endl;
```

```
    return 0;
```

```
}
```

输入:1 2 3 4

输出: \_\_\_\_\_



38 (3分) #include <iostream>

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int n, m;
```

```
    cin >> n >> m;
```

```
    int x = 1, y = 1;
```

```
    int dx = 1, dy = 1;
```

```
    while (true)
```

```
    {
```

```
        int cnt = 0;
```

```
        x += dx;
```

```
        y += dy;
```

```
        if (x == 1 || x == n)
```

```
        {
```

```
            dx = -dx;
```

```
            cnt++;
```

```
        }
```

```
        if (y == 1 || y == m)
```

```
        {
```

```
            dy = -dy;
```

```
            cnt++;
```

```
        }
```

```
        if (cnt == 2)
```

```
        {
```

```
            break;
```

```
        }
```

```
    }
```

```
    cout << x << ", " << y << endl;
```



```
return 0;  
}
```

输入1:4 3

输出1: \_\_\_\_\_

39 (3分) #include <iostream>

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int n, m;
```

```
    cin >> n >> m;
```

```
    int x = 1, y = 1;
```

```
    int dx = 1, dy = 1;
```

```
    while (true)
```

```
    {
```

```
        int cnt = 0;
```

```
        x += dx;
```

```
        y += dy;
```

```
        if (x == 1 || x == n)
```

```
        {
```

```
            dx = -dx;
```

```
            cnt++;
```

```
        }
```

```
        if (y == 1 || y == m)
```

```
        {
```

```
            dy = -dy;
```

```
            cnt++;
```

```
        }
```



```
        if (cnt == 2)
        {
            break;
        }
    }
    cout << x << "," << y << endl;
    return 0;
}
```

输入2:2023 1012

输出2: \_\_\_\_\_

40 (3分) #include <iostream>

using namespace std;

int main()

{

int n, m;

cin >> n >> m;

int x = 1, y = 1;

int dx = 1, dy = 1;

while (true)

{

int cnt = 0;

x += dx;

y += dy;

if (x == 1 || x == n)

{

dx = -dx;

cnt++;

}



```
if (y == 1 || y == m)
{
    dy = -dy;
    cnt++;
}
if (cnt == 2)
{
    break;
}
}
cout << x << "," << y << endl;
return 0;
}
```

输入:2023 1013

输出: \_\_\_\_\_