

# 08 - 多重循环

## C++ 程序设计基础

SOJ 信息学竞赛教练组

2024 年 7 月 20 日

# 目录

1 复习回顾

2 多重循环

3 多重循环的应用

4 break 和 continue 语句

5 总结

# 问题回顾 - 加减数列求和

## 编程题

- 输入一个整数  $n$  ( $1 \leq n \leq 1000$ ), 输出  $1 - 2 + 3 - 4 + 5 - \dots$  一直到  $n$  的和。
- 样例输入  
7
- 样例输出  
4
- 样例说明  
 $1 - 2 + 3 - 4 + 5 - 6 + 7 = 4$

# 问题回顾 - 加减数列求和

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int sum = 0;
9     for (int i = 1; i <= n; i++) {
10         if (i % 2 == 0) {
11             sum -= i;
12         } else {
13             sum += i;
14         }
15     }
16     cout << sum << endl;
17
18     return 0;
19 }
```

既然循环体中可以写顺序结构和分支结构，  
那么是否可以写循环结构呢？

# 目录

1 复习回顾

**2 多重循环**

3 多重循环的应用

4 break 和 continue 语句

5 总结

# 多重循环

- 一重循环

# 多重循环

- 一重循环
- 二重循环



# 多重循环

- 一重循环
- 二重循环
  - 在一重循环的循环体中再写一个循环

# 多重循环

- 一重循环
- 二重循环
  - 在一重循环的循环体中再写一个循环
- 三重循环

# 多重循环

- 一重循环
- 二重循环
  - 在一重循环的循环体中再写一个循环
- 三重循环
- ...

## 例 8.1.1: 输出一行 $m$ 个星号

### 编程题

- 输入一个整数  $m$  ( $1 \leq m \leq 1000$ ), 输出一行  $m$  个星号。
- 样例输入  
5
- 样例输出  
\*\*\*\*\*

## 例 8.1.1: 输出一行 m 个星号

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

## 例 8.1.2: 输出 $n$ 行 $m$ 个星号

### 编程题

- 输入两个整数  $n$  和  $m$  ( $1 \leq n, m \leq 1000$ ), 输出  $n$  行  $m$  个星号。
- 样例输入  
3 5
- 样例输出  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1



# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 1

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 1

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 1

- 输出

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 2

- 输出

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 2

- 输出

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 2

- 输出

\* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 3

- 输出

\* \*



# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 3

- 输出

\* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 3

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j = 4

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

j <= 3 不成立

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 1

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 1

- 输出

\* \* \*



# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 1

- 输出

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 1

- 输出

\* \* \*

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 2

- 输出

\* \* \*

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 2

- 输出

\* \* \*

\*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 2

- 输出

\* \* \*

\* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 3

- 输出

\* \* \*

\* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 3

- 输出

\* \* \*

\* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 3

- 输出

\* \* \*

\* \* \*



# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j = 4

- 输出

\* \* \*

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

j <= 3 不成立

- 输出

\* \* \*

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 2

- 输出

\* \* \*

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

i = 3

- 输出

\* \* \*

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

$i \leq 2$  不成立

- 输出

\* \* \*

\* \* \*

# 示例：双重循环

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "*";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

- 输入

2 3

- 变量

- 输出

\* \* \*

\* \* \*

# 目录

1 复习回顾

2 多重循环

**3 多重循环的应用**

4 break 和 continue 语句

5 总结

## 例 8.2: 输出 $n$ 行从 1 到 $m$ 的整数

### 编程题

- 输入两个整数  $n$  和  $m$  ( $1 \leq n, m \leq 1000$ ), 输出  $n$  行从 1 到  $m$  的整数。
- 样例输入  
2 4
- 样例输出  
1 2 3 4  
1 2 3 4



## 例 8.2：输出 n 行从 1 到 m 的整数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << j << " ";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

## 例 8.3：输出二重循环的循环变量

### 编程题

- 在【例 8.2】的基础上，输出二重循环的循环变量  $(i,j)$  的值，以空格间隔。
- 样例输入  
2 4
- 样例输出  
(1,1) (1,2) (1,3) (1,4)  
(2,1) (2,2) (2,3) (2,4)

## 例 8.3：输出二重循环的循环变量

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             cout << "(" << i << " " << j << ") ";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

## 例 8.3：输出二重循环的循环变量

- 当输入  $n$  为 3,  $m$  为 4 时, 输出如图所示

3 4

(1,1) (1,2) (1,3) (1,4)

(2,1) (2,2) (2,3) (2,4)

(3,1) (3,2) (3,3) (3,4)

## 例 8.3：输出二重循环的循环变量

- 当输入 n 为 3, m 为 4 时, 输出如图所示

3 4

(1,1) (1,2) (1,3) (1,4)  $i = 1$

(2,1) (2,2) (2,3) (2,4)

(3,1) (3,2) (3,3) (3,4)

- i 表示所在的行号

## 例 8.3：输出二重循环的循环变量

- 当输入 n 为 3, m 为 4 时, 输出如图所示

3 4

(1,1) (1,2) (1,3) (1,4)

(2,1) (2,2) (2,3) (2,4)  $i = 2$

(3,1) (3,2) (3,3) (3,4)

- i 表示所在的行号

## 例 8.3：输出二重循环的循环变量

- 当输入 n 为 3, m 为 4 时, 输出如图所示

3 4

(1,1) (1,2) (1,3) (1,4)

(2,1) (2,2) (2,3) (2,4)

(3,1) (3,2) (3,3) (3,4)  $i = 3$

- i 表示所在的行号

## 例 8.3：输出二重循环的循环变量

- 当输入  $n$  为 3,  $m$  为 4 时, 输出如图所示

3 4

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

$j = 1$

- $i$  表示所在的行号
- $j$  表示所在的列号



## 例 8.3：输出二重循环的循环变量

- 当输入  $n$  为 3,  $m$  为 4 时, 输出如图所示

3 4

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

$j = 2$

- $i$  表示所在的行号
- $j$  表示所在的列号

## 例 8.3：输出二重循环的循环变量

- 当输入  $n$  为 3,  $m$  为 4 时, 输出如图所示

3 4

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

$j = 3$

- $i$  表示所在的行号
- $j$  表示所在的列号

## 例 8.3：输出二重循环的循环变量

- 当输入 n 为 3, m 为 4 时, 输出如图所示

3 4

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

j = 4

- i 表示所在的行号
- j 表示所在的列号

## 例 8.4：输出数字直角三角形

### 编程题

- 输入一个整数  $n$  ( $1 \leq n \leq 1000$ ), 输出一个等腰直角三角形, 腰长为  $n$ , 每行由整数 1, 2, 3... 组成, 以空格间隔。
- 样例输入  
3
- 样例输出  
1  
1 2  
1 2 3

## 例 8.4：输出数字直角三角形

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= i; j++) {
10             cout << j << " ";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

## 例 8.4：输出数字直角三角形

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= i; j++) {
10             cout << j << " ";
11         }
12         cout << endl;
13     }
14
15     return 0;
16 }
```

## 例 8.5: 累加数列求和

### 编程题

- 已知  $A_i = 1 + 2 + 3 + \dots + i$ ,  $S_n = A_1 + A_2 + A_3 + \dots + A_n$ 。  
输入一个整数  $n$  ( $1 \leq n \leq 1000$ ), 计算  $S_n$  的结果并输出。
- 样例输入  
3
- 样例输出  
10
- 样例说明  
 $A_1 = 1$   
 $A_2 = 1 + 2 = 3$   
 $A_3 = 1 + 2 + 3 = 6$   
 $S_3 = A_1 + A_2 + A_3 = 1 + 3 + 6 = 10$

## 例 8.5：累加数列求和

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     long long Sn = 0;
9     for (int i = 1; i <= n; i++) {
10         int Ai = 0; // 用于累加 1~i 的累加器，需要初始化为 0
11         for (int j = 1; j <= i; j++) {
12             Ai += j;
13         }
14         Sn += Ai; // 将求得的 Ai 累加到 Sn 中
15     }
16     cout << Sn << endl;
17
18     return 0;
19 }
```



## 例 8.5：累加数列求和

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     long long Sn = 0;
9     for (int i = 1; i <= n; i++) {
10         int Ai = 0; // 用于累加 1~i 的累加器，需要初始化为 0
11         for (int j = 1; j <= i; j++) {
12             Ai += j;
13         }
14         Sn += Ai; // 将求得的 Ai 累加到 Sn 中
15     }
16     cout << Sn << endl;
17
18     return 0;
19 }
```

## 例 8.5：累加数列求和 - 代码优化

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     long long Sn = 0, Ai = 0;
9     for (int i = 1; i <= n; i++) {
10         Ai += i; // 当前的 Ai 为之前的 Ai + i
11         Sn += Ai; // 将求得的 Ai 累加到 Sn 中
12     }
13     cout << Sn << endl;
14
15     return 0;
16 }
```

## 例 8.6: 空心矩形

### 编程题

- 输入两个整数  $n$  和  $m$  ( $1 \leq n, m \leq 1000$ ), 输出  $n$  行  $m$  列的由空格与星号组成的空心矩形。

- 样例输入

4 7

- 样例输出

\*\*\*\*\*

\*       \*

\*       \*

\*\*\*\*\*

## 例 8.6: 空心矩形

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10             if (i == 1 || i == n || j == 1 || j == m) {
11                 cout << "*";
12             } else {
13                 cout << " ";
14             }
15         }
16         cout << endl;
17     }
18
19     return 0;
20 }
```

## 例 8.6: 空心矩形

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n, m;
7     cin >> n >> m;
8     for (int i = 1; i <= n; i++) {
9         for (int j = 1; j <= m; j++) {
10            if (i == 1 || i == n || j == 1 || j == m) {
11                cout << "*";
12            } else {
13                cout << " ";
14            }
15        }
16        cout << endl;
17    }
18    return 0;
19 }
20 }
```

## 例 8.7: 等腰三角形

### 编程题

- 输入一个整数  $n$  ( $1 \leq n \leq 20$ ), 输出高为  $n$  的由空格与星号组成的等腰三角形。

- 样例输入

3

- 样例输出

\*

\*\*\*

\*\*\*\*\*

## 例 8.7：等腰三角形

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int k = n - 1, x = 1; // k 表示空格数量, x 表示星号数量
9     for (int i = 1; i <= n; i++) {
10         for (int j = 1; j <= k; j++) { // 输出 k 个空格
11             cout << " ";
12         }
13         for (int j = 1; j <= x; j++) { // 输出 x 个星号
14             cout << "*";
15         }
16         cout << endl;
17         k--; // 下一行的空格数 - 1
18         x += 2; // 下一行的星号数 + 2
19     }
20
21     return 0;
22 }
```

## 例 8.8：求一个整数中含有多少个 1

### 编程题

- 输入一个整数  $n$  ( $1 \leq n \leq 10^6$ ), 输出  $n$  及其数位中 1 的个数, 用空格间隔。
- 样例输入  
1351
- 样例输出  
1351 2



## 例 8.8：求一个整数中含有多少个 1

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     do {
10         if (n % 10 == 1) {
11             cnt++;
12         }
13         n /= 10;
14     } while (n != 0);
15     cout << n << " " << cnt << endl;
16
17     return 0;
18 }
```

## 例 8.8：求一个整数中含有多少个 1

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     do {
10         if (n % 10 == 1) {
11             cnt++;
12         }
13         n /= 10;
14     } while (n != 0);
15     cout << n << " " << cnt << endl;
16
17     return 0;
18 }
```

- 运行结果正确吗？

## 例 8.8：求一个整数中含有多少个 1

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     do {
10         if (n % 10 == 1) {
11             cnt++;
12         }
13         n /= 10;
14     } while (n != 0);
15     cout << n << " " << cnt << endl;
16
17     return 0;
18 }
```

- 运行结果正确吗？
- 不正确，数位拆分后  $n$  的值会变为 0

## 例 8.8：求一个整数中含有多少个 1

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0, tmp = n;
9     do {
10         if (tmp % 10 == 1) {
11             cnt++;
12         }
13         tmp /= 10;
14     } while (tmp != 0);
15     cout << n << " " << cnt << endl;
16
17     return 0;
18 }
```

## 例 8.8：求一个整数中含有多少个 1

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0, tmp = n;
9     do {
10         if (tmp % 10 == 1) {
11             cnt++;
12         }
13         tmp /= 10;
14     } while (tmp != 0);
15     cout << n << " " << cnt << endl;
16
17     return 0;
18 }
```

- 由于后续需要输出  $n$ ，因此不能直接用  $n$  进行数位拆分

## 例 8.9: 数 1 的个数

### 编程题

- 输入一个整数  $n$  ( $1 \leq n \leq 10^6$ ), 统计从 1 到  $n$  的所有整数中, 数位上 1 的个数。
- 样例输入  
12
- 样例输出  
5
- 样例说明  
当  $n = 12$  时, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 中, 数位上 1 的个数为 5。

## 例 8.9: 数 1 的个数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     for (int i = 1; i <= n; i++) {
10         do {
11             if (i % 10 == 1) {
12                 cnt++;
13             }
14             i /= 10;
15         } while (i != 0);
16     }
17     cout << cnt << endl;
18
19     return 0;
20 }
```

## 例 8.9: 数 1 的个数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     for (int i = 1; i <= n; i++) {
10         do {
11             if (i % 10 == 1) {
12                 cnt++;
13             }
14             i /= 10;
15         } while (i != 0);
16     }
17     cout << cnt << endl;
18
19     return 0;
20 }
```

- 运行结果正确吗?



## 例 8.9: 数 1 的个数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     for (int i = 1; i <= n; i++) {
10         do {
11             if (i % 10 == 1) {
12                 cnt++;
13             }
14             i /= 10;
15         } while (i != 0);
16     }
17     cout << cnt << endl;
18
19     return 0;
20 }
```

- 运行结果正确吗?
- 不正确, 会死循环。  
因为数位拆分后  $i$  会变化!

## 例 8.9: 数 1 的个数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     for (int i = 1; i <= n; i++) {
10         int tmp = i;
11         do {
12             if (tmp % 10 == 1) {
13                 cnt++;
14             }
15             tmp /= 10;
16         } while (tmp != 0);
17     }
18     cout << cnt << endl;
19
20     return 0;
21 }
```

## 例 8.9: 数 1 的个数

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     int n;
7     cin >> n;
8     int cnt = 0;
9     for (int i = 1; i <= n; i++) {
10         int tmp = i;
11         do {
12             if (tmp % 10 == 1) {
13                 cnt++;
14             }
15             tmp /= 10;
16         } while (tmp != 0);
17     }
18     cout << cnt << endl;
19
20     return 0;
21 }
```

- 为了避免程序进入死循环，使用临时变量 tmp 进行数位拆分

# 目录

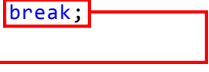
- 1 复习回顾
- 2 多重循环
- 3 多重循环的应用
- 4 break 和 continue 语句**
- 5 总结

# break 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             break;  
5         }  
6     }  
7     ...  
8 }
```

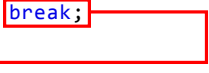
# break 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             break;  
5         }  
6     }  
7     ...  
8 }
```



# break 在多重循环中的应用

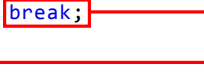
```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             break;  
5         }  
6     }  
7     ...  
8 }
```



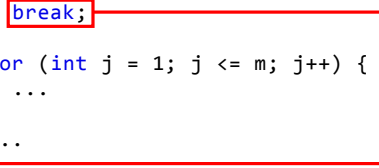
```
1 for (int i = 1; i <= n; i++) {  
2     if (...) {  
3         break;  
4     }  
5     for (int j = 1; j <= m; j++) {  
6         ...  
7     }  
8     ...  
9 }
```

# break 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             break;  
5         }  
6     }  
7     ...  
8 }
```

A red rectangle highlights the 'break;' statement on line 4. A red arrow points from this rectangle to the closing brace of the inner loop on line 6, indicating that the break statement exits the inner loop.

```
1 for (int i = 1; i <= n; i++) {  
2     if (...) {  
3         break;  
4     }  
5     for (int j = 1; j <= m; j++) {  
6         ...  
7     }  
8     ...  
9 }
```

A red rectangle highlights the 'break;' statement on line 3. A red arrow points from this rectangle to the closing brace of the outer loop on line 9, indicating that the break statement exits the outer loop.

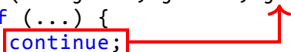


# continue 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             continue;  
5         }  
6     }  
7     ...  
8 }
```

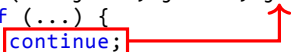
# continue 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             continue;  
5         }  
6     }  
7     ...  
8 }
```



# continue 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             continue;  
5         }  
6     }  
7     ...  
8 }  
  
1 for (int i = 1; i <= n; i++) {  
2     if (...) {  
3         continue;  
4     }  
5     for (int j = 1; j <= m; j++) {  
6         ...  
7     }  
8     ...  
9 }
```



# continue 在多重循环中的应用

```
1 for (int i = 1; i <= n; i++) {  
2     for (int j = 1; j <= m; j++) {  
3         if (...) {  
4             continue;  
5         }  
6     }  
7     ...  
8 }  
  
1 for (int i = 1; i <= n; i++) {  
2     if (...) {  
3         continue;  
4     }  
5     for (int j = 1; j <= m; j++) {  
6         ...  
7     }  
8     ...  
9 }
```

# 目录

- 1 复习回顾
- 2 多重循环
- 3 多重循环的应用
- 4 break 和 continue 语句
- 5 总结

- 多重循环的概念
- 多重循环的应用
  - 输出图形问题
  - 数列计算问题
  - 数位拆分问题

# Thank you!