

集训课-08：双重for循环结构

课程目标

01

打印长方形

02

打印直角三角形

03

打印等腰三角形

04

打印倒立三角形

05

总结回顾



Part 1

打印长方形

打印长方形

每次只可以输出一个*，如何输出***?

打印长方形

```
for (int j = 1; j <= 3; j++)  
{  
    cout << "*";  
}
```

打印长方形

每次只可以输出一个*，如何下面图形？

打印长方形

双重循环结构框架

```
for (int i = 1; i <= n; i++)  
{  
    for (int j = 1; j <= m; j++)  
    {  
    }  
}
```

打印长方形

双重循环结构执行逻辑：

- 1.外层循环执行一次，内层循环执行一轮
- 2.外层循环控制行数，内层循环控制列数

```
for (int i = 1; i <= n; i++)  
{  
    for (int j = 1; j <= m; j++)  
    {  
    }  
}
```

打印长方形

```
for (int i = 1; i <= 3; i++)  
{  
    for (int j = 1; j <= 3; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

打印长方形

想一想1

打印下图的形状，则两个?处应填的数字为？

```
for (int i = 1; i <= ?; i++)  
{  
    for (int j = 1; j <= ?; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```



Part 2

打印直角三角形

打印直角三角形

每次只可以输出一个*，如何输出下面图形？

*

**

打印直角三角形

4 行

*
**

?

列

打印直角三角形

4 行 {
*
**

} ? 列

```
for (int i = 1; i <= 4; i++)
```

```
{
```

```
    for (int j = 1; ? ; j++)
```

```
    {
```

```
        cout << "*";
```

```
    }
```

```
    cout << endl;
```

```
}
```

打印直角三角形

	行号: i	*个数 j
*	第 1 行	1 个
**	第 2 行	2 个
***	第 3 行	3 个
****	第 4 行	4 个

行号i与列数j什么关系？

打印直角三角形

*

**

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= i; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

第 i 行, 打印 i 个 *, 1 个 endl

打印直角三角形

每次只可以输出一个*，如何输出下面图形？

*

打印直角三角形

*

行号: i

第 1 行

第 2 行

第 3 行

第 4 行

*个数 j

1 个

3 个

5 个

7 个

第 i 行, 打印 $2*i-1$ 个 *, 1 个
endl

打印直角三角形

```
*  
***  
*****  
*****
```

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= 2 * i - 1; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

第 i 行，打印 $2*i-1$ 个 $*$ ，1 个 $endl$



Part 3

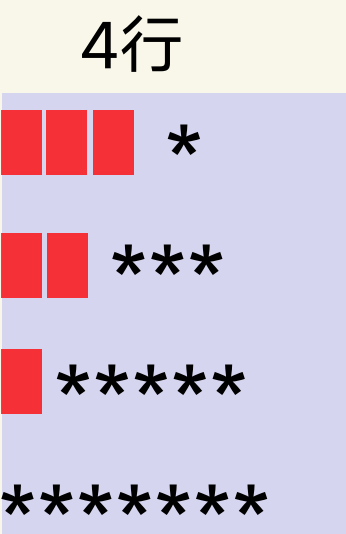
打印等腰三角形

打印等腰三角形

每次只可以输出一个*，如何输出下面图形？

*

打印等腰三角形



行号: i	空格个数
第 1 行	4-1
第 2 行	4-2
第 3 行	4-3
第 4 行	4-4

第 i 行，打印 $4 - i$ 个空格

打印等腰三角形

4行

```
  *  
 ***  
*****  
*****
```

行号: i

*个数

第 1 行

1 个

第 2 行

3 个

第 3 行

5 个

第 4 行

7 个

第 i 行, 打印 $2 * i - 1$ 个*

打印等腰三角形

4行

```
  *
 ***
*****
*****
```

```
for (int i = 1; i <= 4; i++)
{
    for (int j = 1; j <= 4 - i; j++)
    {
        cout << " ";
    }

    for (int k = 1; k <= 2 * i - 1; k++)
    {
        cout << "*";
    }

    cout << endl;
}
```



Part 4

打印倒立三角形

打印倒立三角形

每次只可以输出一个*，如何输出下面图形？

*

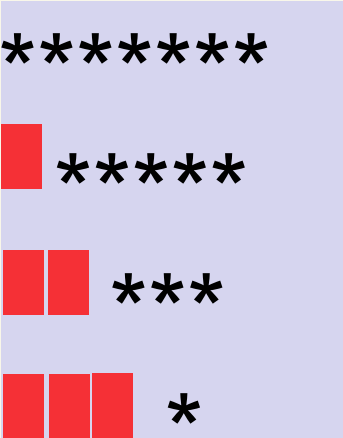
打印倒立三角形

4行	行号: i	空格个数j
*****	第 1 行	0个
*****	第 2 行	1 个
***	第 3 行	2 个
*	第 4 行	3 个

第 i 行，打印 i - 1 个空格

打印倒立三角形

4行



```
*****
 *
 *
 *
 *
```

行号: i

*个数 k

第 1 行

7 个

第 2 行

5 个

第 3 行

3 个

第 4 行

1 个

第 i 行, 打印 $2 * (4 - i) + 1$ 个*

打印倒立三角形

*

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= i - 1; j++)  
    {  
        cout << " ";  
    }  
  
    for (int k = 1; k <= 2 * (4 - i) + 1; k++)  
    {  
        cout << "*";  
    }  
  
    cout << endl;  
}
```



Part 5

总结回顾

总结回顾

双重循环结构框架

```
for (int i = 1; i <= n; i++)  
{  
    for (int j = 1; j <= m; j++)  
    {  
    }  
}
```

总结回顾

双重循环结构执行逻辑：

- 1.外层循环执行一次，内层循环执行一轮
- 2.外层循环控制行数，内层循环控制列数

```
for (int i = 1; i <= n; i++)  
{  
    for (int j = 1; j <= m; j++)  
    {  
    }  
}
```

总结回顾

```
for (int i = 1; i <= 3; i++)  
{  
    for (int j = 1; j <= 3; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

总结回顾

*

**

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= i; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

第 i 行, 打印 i 个 *, 1 个 endl

总结回顾

```
*  
***  
*****  
*****
```

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= 2 * i - 1; j++)  
    {  
        cout << "*";  
    }  
    cout << endl;  
}
```

第 i 行，打印 $2*i-1$ 个 $*$ ，1 个 `endl`

总结回顾

4行

```
  *  
 ***  
*****  
*****
```

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= 4 - i; j++)  
    {  
        cout << " ";  
    }  
  
    for (int k = 1; k <= 2 * i - 1; k++)  
    {  
        cout << "*";  
    }  
  
    cout << endl;  
}
```

总结回顾

4行

*

```
for (int i = 1; i <= 4; i++)  
{  
    for (int j = 1; j <= i - 1; j++)  
    {  
        cout << " ";  
    }  
  
    for (int k = 1; k <= 2 * (4 - i) + 1; k++)  
    {  
        cout << "*";  
    }  
  
    cout << endl;  
}
```

A horizontal yellow banner with blue triangular ends, centered on a solid blue background. The text "Class is over" is written in blue within the banner.

Class is over