

三大循环

while 语法结构

```
语法  
while(条件表达式)  
  
    循环语句;
```

执行顺序：

- ①先对条件表达式进行判断（条件满足时，执行循环；不满足时，终止循环）
- ②执行循环语句。
- ③执行完所有循环语句，再进行条件判断。

注意：单条循环语句可以不加花括号{}，如有多条循环语句，得加上花括号{}。

for循环

```
语法  
for(表达式1; 表达式2; 表达式3)  
  
    循环语句;
```

表达式1

表达式1为**初始化部分**，用于初始化循环变量的。

表达式2

表达式2为条件判断部分，用于判断循环时候终止。

表达式3

表达式3为**调整部分**，用于循环条件的调整。

执行顺序：

- ①初始化(只会在for一开始的时候执行一次)
- ②条件判断
- ③执行循环语句
- ④执行完循环语句后，进行调整部分，然后进行条件判断。

注意：单条循环语句可以不加花括号{}，如有多条循环语句，得加上花括号{}。

do...while()循环

语法

do

循环语句；

while(条件表达式)；

执行顺序：

①先执行do里面的循环语句

②执行完后，进行条件判断，满足则继续执行do里面的循环语句，不满足就结束循环。

注意：单条循环语句可以不加花括号{}，如有多条循环语句，得加上花括号{}。

基本类型

1、整型

int：基本整数型

short：短整型

long：长整型

long long：双长整型

2、字符型

char：字符型，用于存储单个字符，占1个字节。

3、浮点型

float：单精度浮点型

double：双精度浮点型

4、布尔型 (c++才有)

bool

switch

switch基本语法结构如下

```
switch (expression) {  
    case constant-expression :  
        statement(s);  
        break; /* 可选的 */  
    case constant-expression :  
        statement(s);  
        break; /* 可选的 */  
    /* 您可以有任意数量的 case 语句 */  
    default : /* 可选的 */  
        statement(s);  
}
```

switch语句必须遵循下面的规则：

- switch 语句中的 **expression** 是一个常量表达式，必须是一个**整型或枚举类型**。
- 在一个 switch 中可以有任意数量的 case 语句。每个 case 后跟一个要比较的值和一个冒号。
- case 的 **constant-expression** 必须与 switch 中的变量具有**相同的数据类型**，且必须是一个常量或字面量。
- 当被测试的变量等于 case 中的常量时，case 后跟的语句将被执行，直到遇到 break 语句为止。
- 当遇到 break 语句时，switch 终止，控制流将跳转到 switch 语句后的下一行。
- 不是每一个 case 都需要包含 break。如果 case 语句不包含 break，控制流将会继续 后续的 case，直到遇到 break 为止。
- 一个 switch 语句可以有一个可选的 default case，出现在 switch 的结尾。default case 可用于在上面所有 case 都不为真时执行一个任务。default case 中的 break 语句不是必需的。