



C++第 11 课 do while 循环

课程讲义



01 //探索新知 //

1 认识do...while

前面的课程中我们学习了while循环，让计算机重复多次的干一件事情，今天我们来认识一下do...while循环。

和 while 循环最明显的区别，就是do...while 循环中，判断是否继续循环的条件，放在后面。也就是说，就算是条件一开始就不成立，循环也要被执行一次。

```
do{  
    要执行的语句;  
} while (循环条件);
```

- (1)先执行一次循环体，然后再判断表达式
- (2)当表达式的值为真（非0）时，重新执行循环体
- (3)一直重复执行，直到表达式的值为假（也就是0），退出循环

实例1：高斯公式

计算 $1+2+3+\cdots+100$ 的和

```
#include <iostream>  
using namespace std;  
int main() {  
    int a = 1, sum = 0;  
    while (a <= 100) {  
        sum += a;  
        a += 1;  
    }  
    cout << sum;  
    return 0;  
}
```

```
#include <iostream>  
using namespace std;  
int main() {  
    int i = 1, sum = 0;  
    do {  
        sum += i;  
        i++;  
    } while (i <= 100);  
    cout << sum << endl;  
    return 0;  
}
```

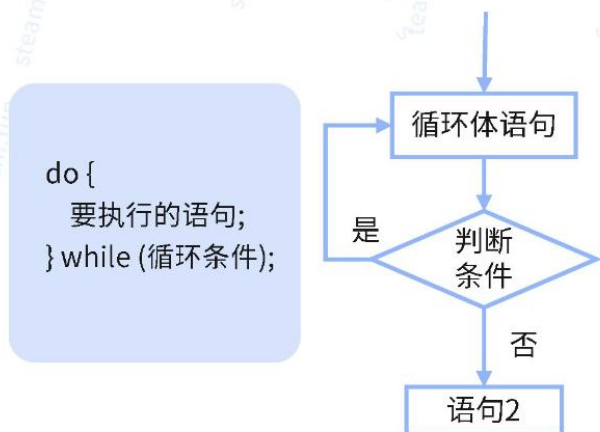


01 // 探索新知 //

2 与while的区别

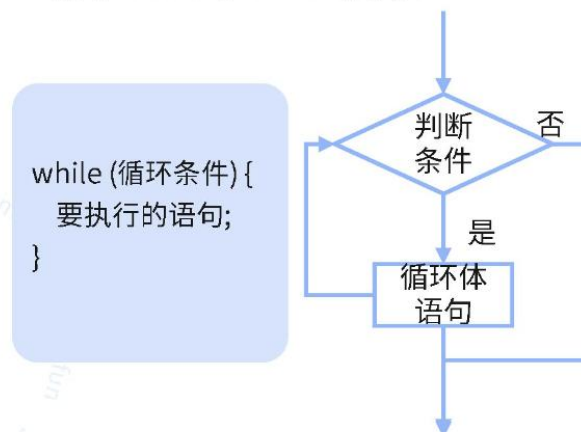
do...while循环:直到型循环

1. 先执行循环体, 再判断条件
2. 循环体至少要执行一次



while循环:当型循环

1. 先判断执行的条件, 再决定是否执行循环体
2. 循环体也有可能一次不执行



同学们想一想, 下面两个例子那个是while循环、哪个是do...while循环

先判断信号灯什么颜色,
然后再决定走不走先抽奖一次, 再进行判断,
没抽到中奖数字继续



02 //强化练习//

1

找bug

用你的火眼金睛找一找下面的程序都有什么样的错误吧，请填写在划线的位置。

```
#include <iostream>
using namespace std;
int main() {
    int a = 10, y = 0;
    do {
        a += 2;
        y += a;
        cout << "a=" << a << ", " << "y=" << y << endl;
    } while(a == 14)
    return 0;
}
```

找bug

```
#include <iostream>
using namespace std;
int main() {
    int i = 1;
    do {
        cout << i;
    } while(i <= 5);
    i++;
    return 0;
}
```

找bug



02 //强化练习//

2

判断结果

```
#include <iostream>
using namespace std;
int main() {
    int i = 1;
    do {
        cout << i;
        i++;
    } while(i <= 5);
    return 0;
}
```

输出

输出: _____

```
#include <iostream>
using namespace std;
int main() {
    int i = 1;
    do {
        i++;
        cout << i;
    } while(i <= 5);
    return 0;
}
```

输出

输出: _____

《练手项目》

随机生成一个 0~100 之间的一个数，输入你要猜测的数。

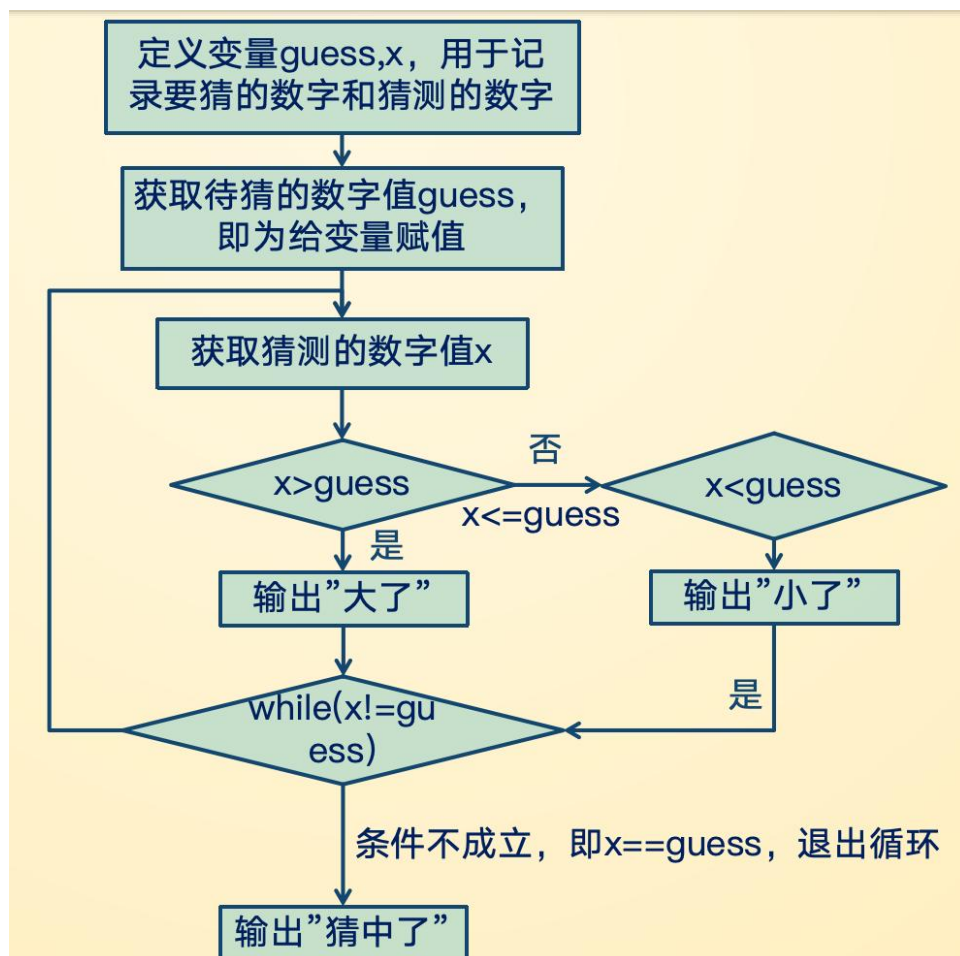
如果输入的数大于初始的数则提示大了，

如果输入的数小于初始的数则提示小了，

如果输入的数等于初始的数则提示相等。

你能否猜出这个数呢？

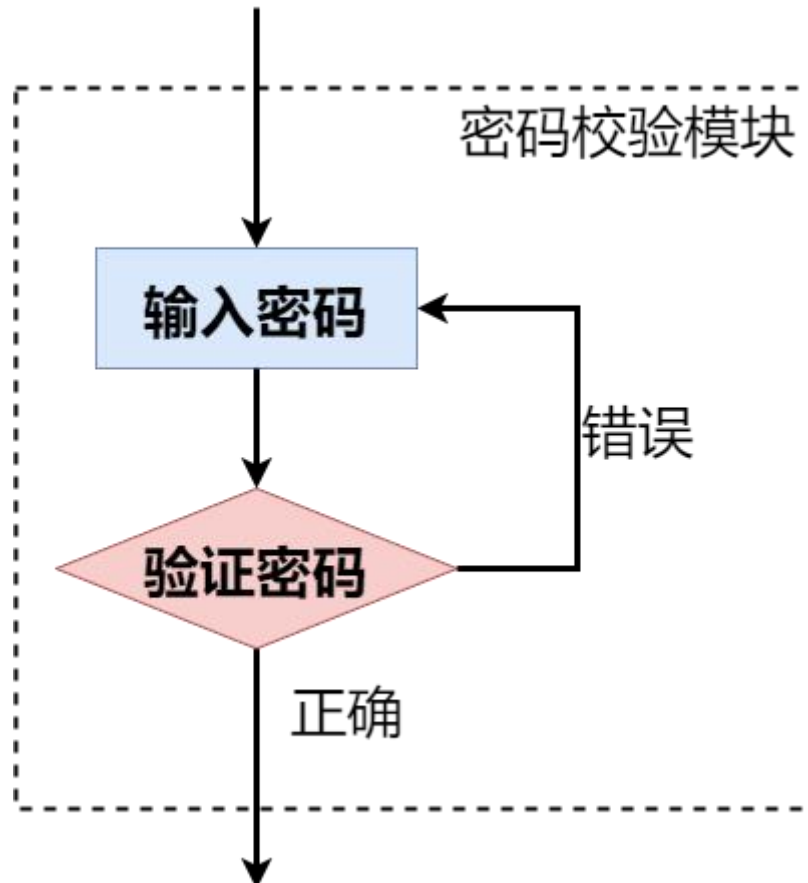
每猜一次，计算机都会告诉你猜的数是大了还是小了，直到你猜出这个数为止。



很多功能完整成熟的应用程序软件都会设计用户登录时的密码校验模块，我们下面运用 do...while...语句设计一个简单的密码校验模块。

模块要求：

1. 初始密码为 123456
2. 如果输入密码和初始密码相同，输出正确。
3. 如果输入密码和初始密码不相同，提示错误，继续输入，直到密码正确才结束输入。



```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int pwd=0;
    do
    {
        cout<<"请输入密码：";
        cin>>pwd;
    }while(pwd!=123456);
    return 0;
}
```

随机数常用于许多不同类型的程序中，以下是一些示例：

- 计算机游戏通常要使用随机数字来模拟一些随机过程，例如掷骰子或发牌。
- 模拟程序使用随机数字来决定后续将要发生的一系列操作或人与动物等的行为
- 可以创建使用随机数字的公式以确定特定事件在程序中出现的时间。
- 数据分析程序可能会使用随机数字随机选择要检验的数据。
- 计算机安全系统使用随机数字来加密敏感数据。

rand() 函数

rand() 函数，每次调用该函数都将返回一个 0~32767 间的非负整数。

该函数返回的数字其实是伪随机数，即具有随机数的表现和属性，但实际上并不是随机的，它们实际上是用算法生成的。

该函数需要一个起始值，称为种子，以生成数字。如果没有给出一个种子，那么它将在每次运行时产生相同的数字流。