



C++

轉向

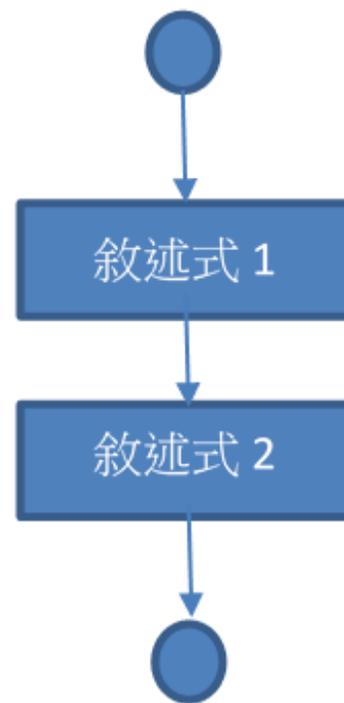




循序結構

循序結構

程式中的敘述式是以他們在程式中的順序一個接一個地執行。





改變循序
的方法

改變循序的方法

C++ 有一些敘述式，可以指定下一個執行的敘述式，這叫做 **控制權的移轉**。



if...goto

if...goto的語法

```
標籤：  
敘述式；  
if(條件式){  
    goto 標籤；  
}
```

goto 一定要與if配合，否則停不下來(無窮迴圈)

```
標籤：  
敘述式；  
// if(條件式){  
    goto 標籤；  
// }
```

問題

HaHaHa!
印10次



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

```
int main(){
```

```
    int i=0;
    repeat:
```

```
    cout<<"HaHaHa!"<<endl;
    i=i+1;
```

```
    if (i<10){
        goto repeat;
    }
```

```
    return 0;
```

```
}
```

重複的開始

重複的開始

重複的終止條件

循序結構與控制權轉移

循序結構

程式中的敘述式，是以在程式中的順序一個一個被執行

控制權轉移

程式中的敘述式，能夠讓程式設計師指定下一個執行的敘述式

Goto

Goto就是控制權的轉移，1960年代，人們發現任意控制權的移轉，將會使得軟體發展越發困難，批評的。因為工程師可以做極大範圍的控制權移轉，偵錯與修改難度較高。

為什麼教？

教你一種計算器很直覺式的思維，讓你瞭解重複結構的發展歷程。

控制權移轉的終結者

- Bohm和Jacopini已經證明程式可以不必使用goto敘述式來撰寫，開始轉變沒有goto的程式設計方法
- 提出結構化程式設計的理論
 - 程式清晰易懂
 - 更容易偵錯
 - 軟體計劃更容易準時完成

結構化程式設計

- 所有的程式皆由三種結構所構成
- 循序結構
- 選擇結構
 - 單一選擇敘述式：if
 - 雙重選擇敘述式：if...else
 - 多選一敘述式：switch
- 重複結構
 - while
 - do ...while
 - for