

重複結構- 固定次數



程式設計的三大結構

循序



判斷



重複



重複

回想過去的問題...

問題：

寫一個程式，輸入一個整數，
如果是偶數，則印出此數是偶數，
如果是奇數則印出此數是奇數。
此程式可執行2次。

輸入範例:

78

77

輸出範例:

78是偶數

77是奇數

```
#做第一次
cin >> num;
if(num%2==0)
{
    cout << num << "是偶數" << endl;
}
else{
    cout << num << "是奇數" << endl;
}
#做第二次
cin >> num;
if(num%2==0)
{
    cout << num << "是偶數" << endl;
}
else{
    cout << num << "是奇數" << endl;
}
```

如果是執行次數是1000次？

- 找出重複的主體

```
cin >> num;
if(num%2==0)
{
    cout << num << "是偶數" << endl;
}
else{
    cout << num << "是奇數" << endl;
}
```

- 加上一個可以有此數的重複

i=1

while (i<=1000){

重複的主體

i=i+1; # 每做一次，次數 + 1

}



再看
被老師罰寫的例子~~

老師下的指令，不可能一直寫，肯定會有停下來
來的條件，**停下來**的指令有2類



寫X次



寫到下課為止

寫一個程式，顯示「HaHaHa!」10次。

輸入範例:

輸出範例:

HaHaHa!

HaHaHa!

HaHaHa!

HaHaHa!

.....

問題



重複結構-while

```
while(條件表達式)
```

作為是否執行重複的**控制條件**表達式，表達式如果成立則執行迴圈。

```
{
```

```
  語句1;
```

```
  語句2;
```

```
  ...
```

重複部分由多個語句構成，應由一對括號括起來，以構成一個語句塊的形式。

```
}
```

HaHaHa寫10次

```
#include<iostream>
using namespace std;
int main()
{
    int i=0;
    while(i<10)
    {
        cout<<"HaHaHa!"<<endl;
        i=i+1;
    }
    return 0;
}
```

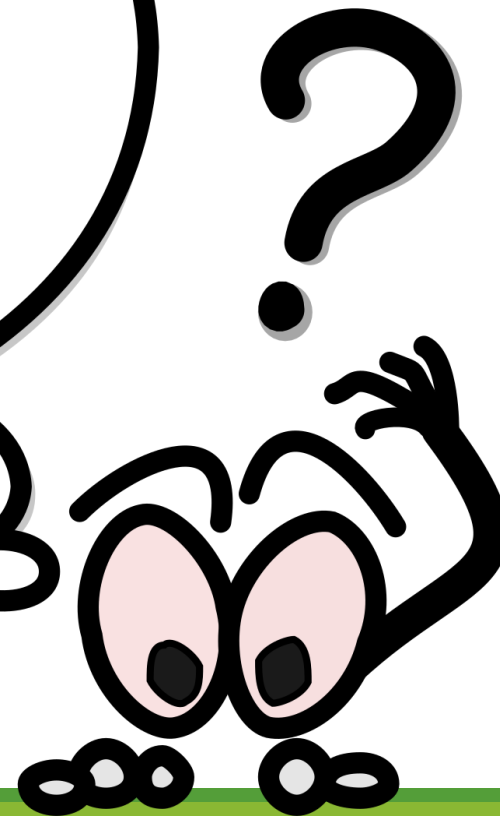
控制條件要記得設定初始值

控制條件表達式，i必須小於10

重複主體由執行主體與條件改變兩個語句所構成

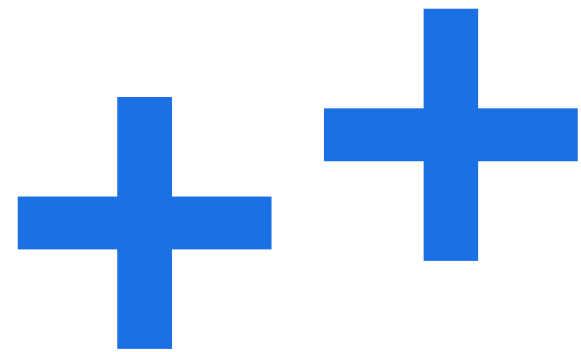
寫一個程式，我說幾次，
就顯示幾次「HaHaHa!」

問題



```
#include<iostream>
using namespace std;
int main()
{
    int i=0;
    int n;
    cin>>n;
    while(i<n){
        cout<<"HaHaHa!"<<endl;
        i=i+1;
    }
    return 0;
}
```

控制條件表達式，i必須小於n



延伸學習



強制轉換

整數除以整數時，
如果想要得到浮點
數的結果

```
average=(float) sum/n;  
cout<<fixed<<setprecision(2)<<average<<"\n";
```

程式寫作風格提醒

```
#include<iostream>
using namespace std;
int main()
{
    int i=0;
    int n;
    cin>>n;
    while(i<n){
        cout<<"HaHaHa!"<<endl;
        i=i+1;
    }
    return 0;
}
```

程式寫作風格
寫迴圈時，迴圈內的語句相對應於while縮兩格