



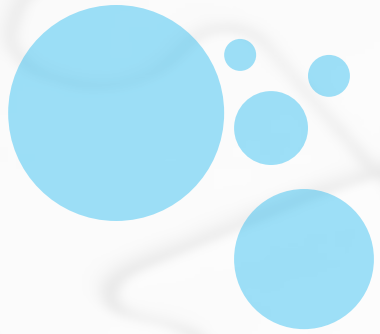
2017

程式設計
加強班

程式設計與實習(一)

BY 孫茂勛

Email:JOHN85051232@GMAIL.COM



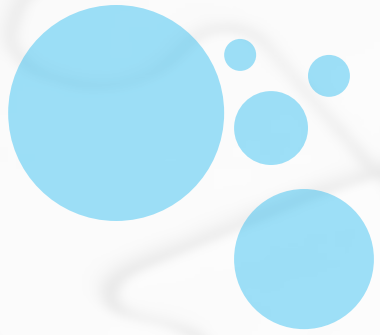
Loop

Q：如何印出Hello World 10次?100次?1000次?....

A：~~複製貼上...~~

迴圈(Loop)

- ❖ 當要電腦做的事情具有某種程度的規律時
- ❖ 縮短程式碼長度
- ❖ C的三種Loop：for、while、do...while



Loop

```
for(exp1 ; exp2 ; exp3)
{
    ...
}
```

exp1 : 區域變數設置

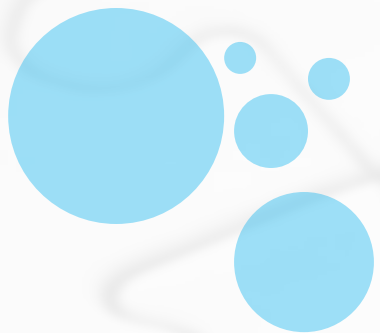
exp2 : 終止條件(跳出迴圈)

exp3 : 每執行完一次後會執行的動作(++ / --)



Loop

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5
6     int num = 0;
7     int check = 0;
8
9     for(num = 1 ; num <= 10 ; num++) {
10         printf("%d\n" , num);
11     }
12     system("pause");
13     return 0;
14 }
15
```



Loop

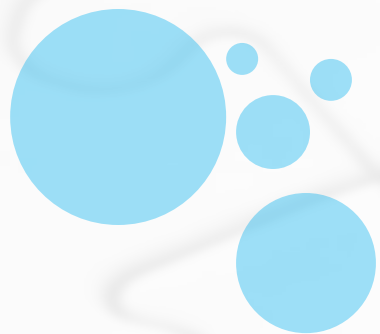
```
while(exp1)
{
    ...
}
```

exp1 : 終止條件

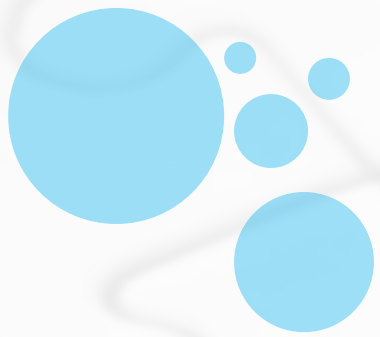
```
while(true){...}
```

break 、 continue

```
int n = 0;
while( true)
{
    printf( "hello world\n" );
    n++; //n = n+1
    if(n == 10)
    {
        break;
    }
}
```



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5
6     int num = 0;
7
8
9     while(1) {
10         num += 1;
11         if(num == 11) {
12             break;
13         }
14         else if(num == 5 || num == 9) {
15             continue;
16         }
17         printf("%d\n" , num);
18     }
19
20     system("pause");
21     return 0;
22 }
```



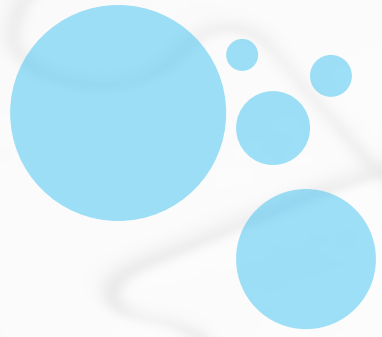
練習

看圖說故事

CH3 P.62


```
1  Set total to zero
2  Set grade counter to one
3
4  While grade counter is less than or equal to ten
5      Input the next grade
6      Add the grade into the total
7      Add one to the grade counter
8
9  Set the class average to the total divided by ten
10 Print the class average
```

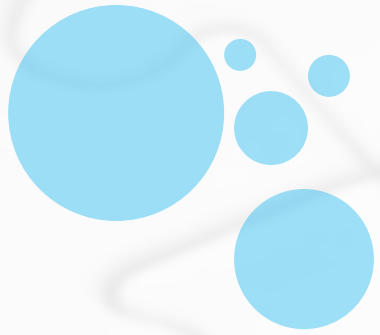
Fig. 3.5 | Pseudocode algorithm that uses counter-controlled iteration to solve the class-average problem.



Formulating Algo with Top-Down...

CH3 P.73

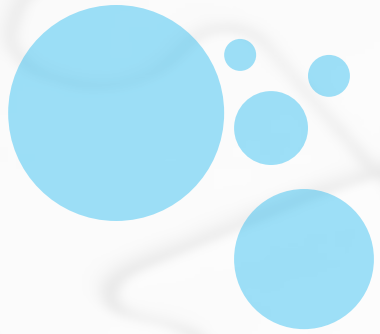
 Top-Down是一種設計程式的**想法**，作者告訴你在寫程式前先了解你到底要用程式做出什麼(明確目的)，然後一步一步的去將細節設計出來.....有那麼簡單就好了。



Type Conversion

➤ Explicit conversion & Implicit conversion

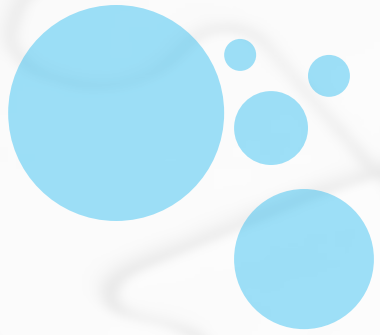
➤ <http://notepad.yehyeh.net/Content/CPP/CH01/03DataType/4.php>



Function

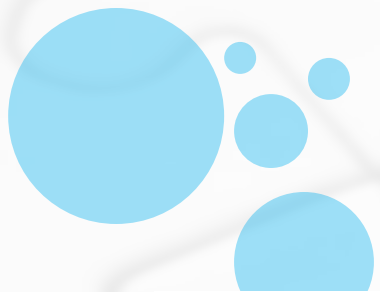
函式(Function)

- ❖ C語言寫程式的模組單位。
- ❖ 將程式切割成不同功能模組的組合。
- ❖ 使程式碼容易閱讀、維護。
- ❖ 架構包含：函式型態、函式名稱、參數、程式碼。
- ❖ 例如：給一台機器（函式型態、名稱）電力（參數），他就可以執行某些工作（功能）。



Function





Function

```
#include <stdio.h>
#include <stdlib.h>

void printf_star(int star_number)
{
    for(int i = star_number ; i > 0 ; i--)
    {
        printf( "*\n");
    }
}

int main()
{
    printf_star(5); //可以直接放入參數
    int n = 10;
    printf_star(n); //也可以放入變數
    system( "PAUSE");
    return 0;
}
```

宣告方式:

函式型態 函數名稱(可能會用到的參數)

```
{
    ...
}
```

當編譯器讀到main中的printf_star時，他會去尋找程式碼中有這一個名字的函數開始執行，執行完才會回到main中繼續執行下一行指令。

Function

```
#include <stdio.h>
#include <stdlib.h>

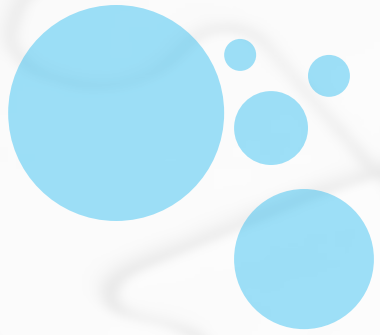
void printf_star(int star_number);

int main()
{
    printf_star(5); //可以直接放入參數
    int n = 10;
    printf_star(n); //也可以放入變數
    system("PAUSE");
    return 0;
}


void printf_star(int star_number)
{
    for(int i = star_number ; i > 0 ; i--)
    {
        printf("*\n");
    }
}
```

函式宣告在`main()`後面時，由於編譯器是由上往下讀，所以會看不懂`main`中的`printf_star`是什麼。

所以要在一開始先加入一個宣告，告訴編譯器這是我們自定義的函式。



練習

 用*印出正三角形、倒三角形

*

* * * *

* *

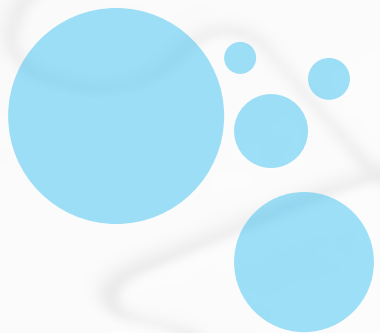
* * *

* * *

* *

* * * *

*



Function

宣告方式:

函式型態 函數名稱(可能會用到的參數)

{

...

回傳值

}

不同類型的函式以及回傳值：

函式類型

void

int

float

char

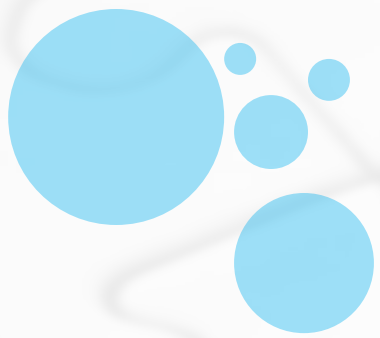
回傳值

不用回傳

return int

return float

return char



Function

```
int add_two(int input1,int input2)//一個將兩數相加的函式，參數是兩個整數
{
    return input1 + input2;//回傳一個整數，他是兩數相加的結果
}
int main( )
{
    int input1 = 0,input2 = 0;
    scanf( "%d %d",&input1,&input2);
    printf( "%d + %d = %d\n",input1,input2);

    system( "PAUSE");
    return 0;
}
```

```
3
99
3 + 99 = 102
請按任意鍵繼續 . . .
```




THANK YOU