



Computer Engineering

วิศวกรรมคอมพิวเตอร์



## บทที่ 5 การเขียนโปรแกรมแบบวนซ้ำ

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

- นักศึกษาเข้าใจกระบวนการทำงานแบบวนซ้ำ
- นักศึกษาสามารถเขียนโปรแกรมภาษาซีโดยใช้คำสั่งเพื่อให้คอมพิวเตอร์ทำงานแบบวนซ้ำได้

- คำสั่งการวนซ้ำด้วย while
- คำสั่งการวนซ้ำด้วย do - while
- คำสั่งการวนซ้ำด้วย for

# 5.1 ทำไมต้องการเขียนโปรแกรมต้องมีการวนซ้ำ



01006012 Computer Programming

- เหตุการณ์ที่เกิดขึ้นหลายรอบ

เช่น โปรแกรมแสดง ชื่อ 20 ครั้ง

- เหตุการณ์ที่เกิดขึ้นหลายรอบ โดยมีการเปลี่ยนแปลงค่า หรือมีเงื่อนไข

เช่น แสดงผลเลข 0, 1, 2, ... , 10

แสดงผลรวมของ 1,3,5,7, ... , 99

แสดง ชื่อ ไปเรื่อย ๆ จนกว่าตัวแปร X จะมีค่ามากกว่า 30

# โปรแกรม 5.1 แสดงผลตัวเลข 0 – 10 | โดยไม่มีการวนซ้ำ

01006012 Computer Programming

จงเขียนผังงานและโปรแกรมเพื่อแสดงตัวเลข 0 - 10 ออกทางหน้าจอ

- Output Analysis
  - แสดงผลเลข 0, 1, 2, ... , 10
- Input Analysis
  - ไม่มี

# โปรแกรม 5.1 แสดงผลตัวเลข 0 – 10 | โดยไม่มีการวนซ้ำ

01006012 Computer Programming

- Process Analysis
  - โปรแกรมแสดงผลเลข 0, 1, 2, ... ,10
- Variable Define
  - ไม่ใช่ (หรือใช้ count เพื่อเพิ่มค่า)

# โปรแกรม 5.1 แสดงผลตัวเลข 0 – 10 | โดยไม่มีการวนซ้ำ



01006012 Computer Programming

```
#include<stdio.h>
#include<conio.h>
int main()
{
    printf  ("0\t");
    printf  ("1\t");
    printf  ("2\t");
    printf  ("3\t");
    printf  ("4\t");
    ...
    ...
    printf  ("10\t");
    return 0;
}
```

```
#include<stdio.h>
#include<conio.h>
int main()
{
    int      count = 0;
    printf  ("%d\t",count++);
    printf  ("%d\t",count++);
    printf  ("%d\t",count++);
    printf  ("%d\t",count++);
    printf  ("%d\t",count++);
    ...
    ...
    printf  ("%d\t",count++);
    return 0;
}
```

# โปรแกรม 5.1 แสดงผลตัวเลข 0 – 10 | โดยใช้คำสั่งวนซ้ำ



01006012 Computer Programming

```
#include<stdio.h>
#include<conio.h>
int main()
{
    int    count = 0;
    while (count <= 10)
    {
        printf ("%d\t",count) ;
        count++ ;
    }

    return 0;
}
```



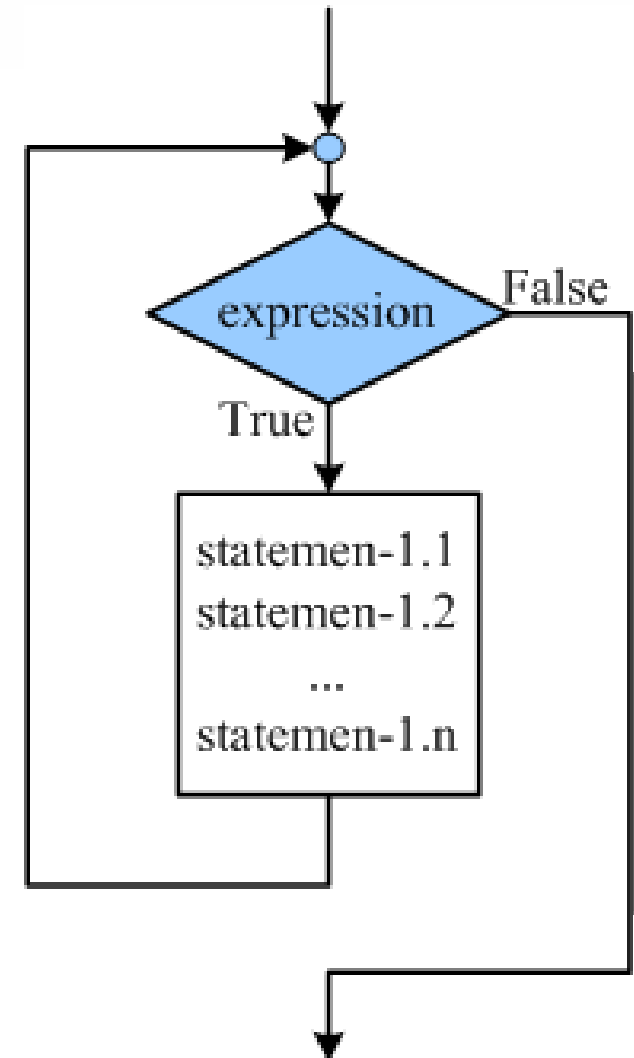
- **while**
- **do - while**
- **for**

## 5.2 คำสั่ง while

01001012 Principle of Computer Programming

```
while (expression)  
  statement-1;
```

```
while (expression)  
{  
  statement-1.1;  
  statement-1.2;  
  ...  
  statement-1.n;  
}
```



# โปรแกรม 5.2 แสดงผลตัวเลข 0 – 10 | while

01006012 Computer Programming

จงเขียนผังงานและโปรแกรมที่มีการควบคุมทิศทางแบบวนรอบ โดยใช้คำสั่ง while เพื่อแสดงตัวเลข 0 - 10 ออกทางหน้าจอ

- Output Analysis
  - แสดงผลเลข 0, 1, 2, ... , 10
- Input Analysis
  - ไม่มี

# โปรแกรม 5.2 แสดงผลตัวเลข 0 – 10 | while

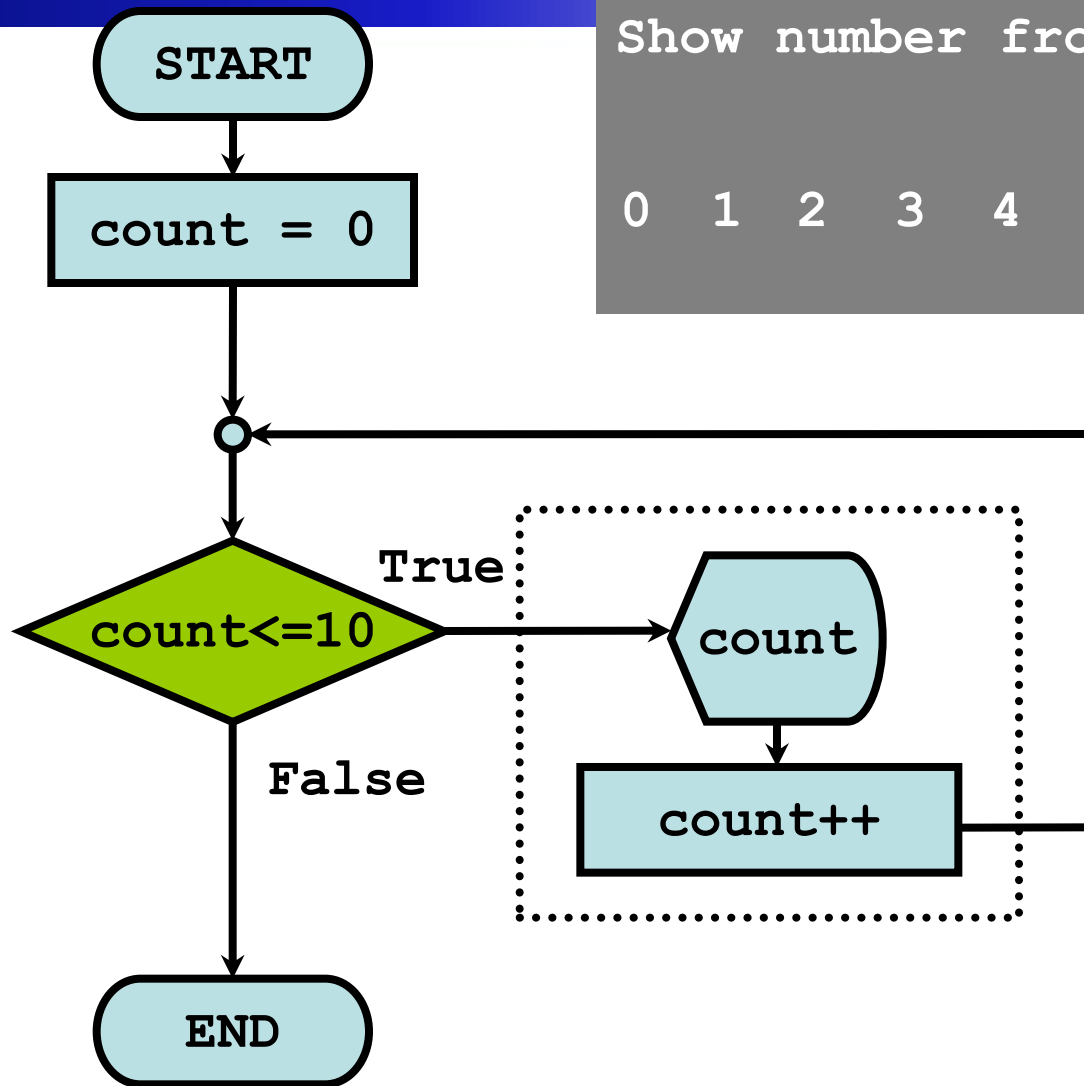
01006012 Computer Programming

- Process Analysis
  - โปรแกรมทำงานแบบวนรอบ เพื่อแสดงผลเลข 0, 1, 2, ... ,10
- Variable Define
  - count เป็นจำนวนเต็มเพื่อใช้นับจำนวนรอบ

count = 11

Show number from zero to ten

0 1 2 3 4 5 6 7 8 9 10



# โปรแกรม 5.2 แสดงผลตัวเลข 0 – 10 | while

01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    int    count = 0;
    printf ("Show number from zero to ten\n\n");
    while (count<=10)
    {
        printf ("%d\t",count);
        count++;
    }
    return 0;
}
```

Show number from zero to ten

0      1      2      3      4      5      6      7      8      9      10

จงเขียนผังงานและโปรแกรมที่มีการควบคุมทิศทางแบบวนรอบ โดยใช้คำสั่ง while เพื่อให้โปรแกรมทำการบวกเลขจำนวนเต็ม ตั้งแต่ 1 จนถึงค่าที่ผู้ใช้งานกำหนด

- Output Analysis
  - ผลลัพธ์การบวกเลขจำนวนเต็ม ตั้งแต่ 1 ถึงค่าที่ผู้ใช้กำหนด
- Input Analysis
  - ค่าที่ผู้ใช้งานป้อนเข้ามา

# โปรแกรม 5.3 แสดงผลตัวเลข 1 ถึงค่าที่กำหนด | while



01006012 Computer Programming

- Process Analysis

- โปรแกรมถามว่าผู้ใช้งานต้องการบวกเลขตั้งแต่ 1 ถึงเลขใด
- วงรอบแบบ while เพื่อบวกค่า
- แสดงผลลัพธ์ที่ได้

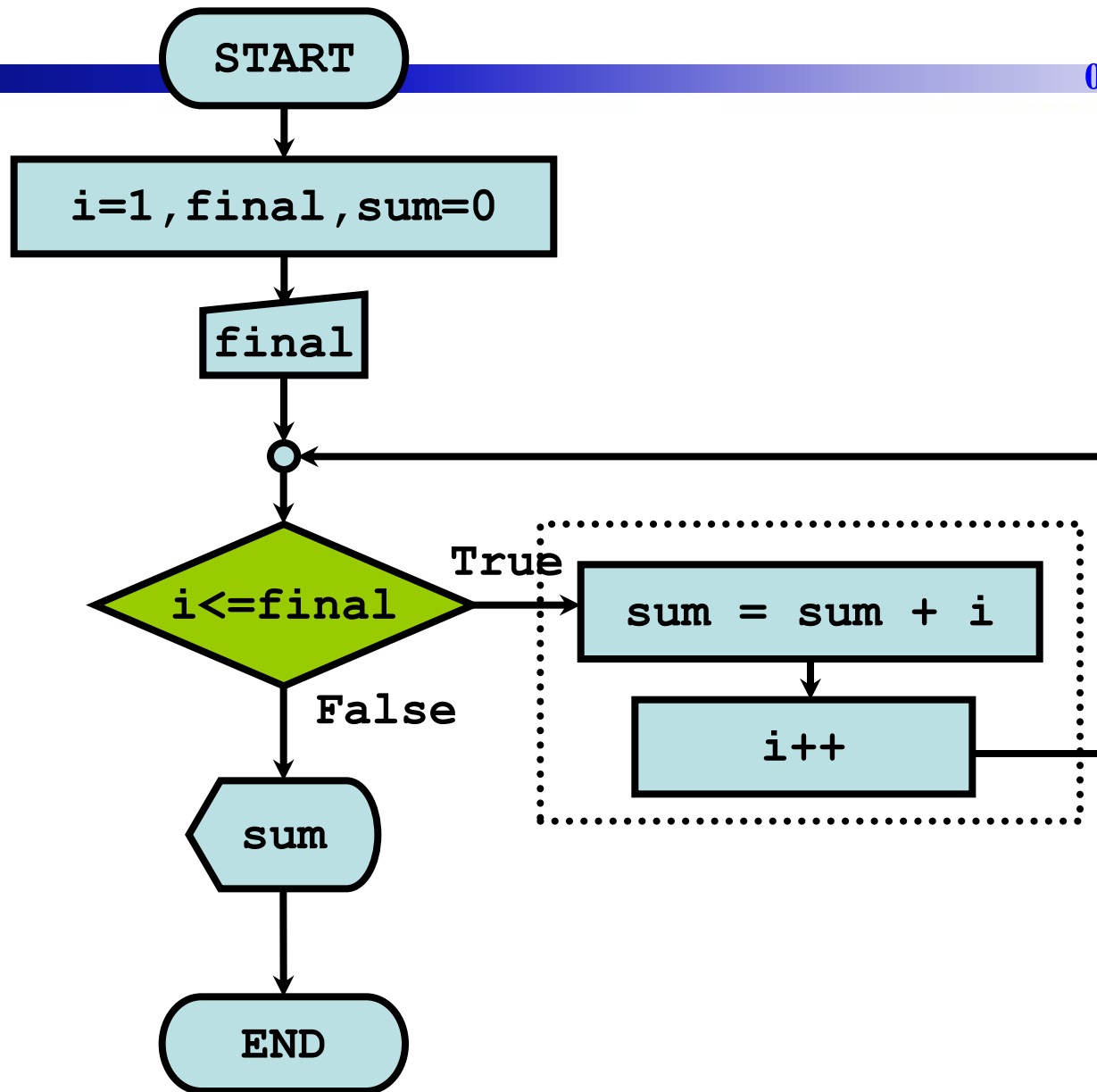
- Variable Define

sum = 0    ผลรวมของการบวก โดยเริ่มต้นมีค่าเท่ากับ 0

i = 1      ค่าที่นำเข้าไปบวกกับ sum ในแต่ละรอบ โดยรอบแรกค่า i  
มีค่าเท่ากับ 1 และมีค่าเพิ่มขึ้นรอบละ 1

final      เพื่อรับค่าจากผู้ใช้ และกำหนดจุดสิ้นสุดของค่า i





# โปรแกรมแสดงผลตัวเลข 1 ถึงค่าที่กำหนด | while



01006012 Computer Programming

```
int main()
{
    int    i = 1, final, sum=0;
    printf ("Enter final number : ");
    scanf  ("%d",&final);
    while (i<=final)
    {
        sum = sum + i;
        i++;
    }
    printf  ("Sum = %d",sum);
    return 0;
}
```

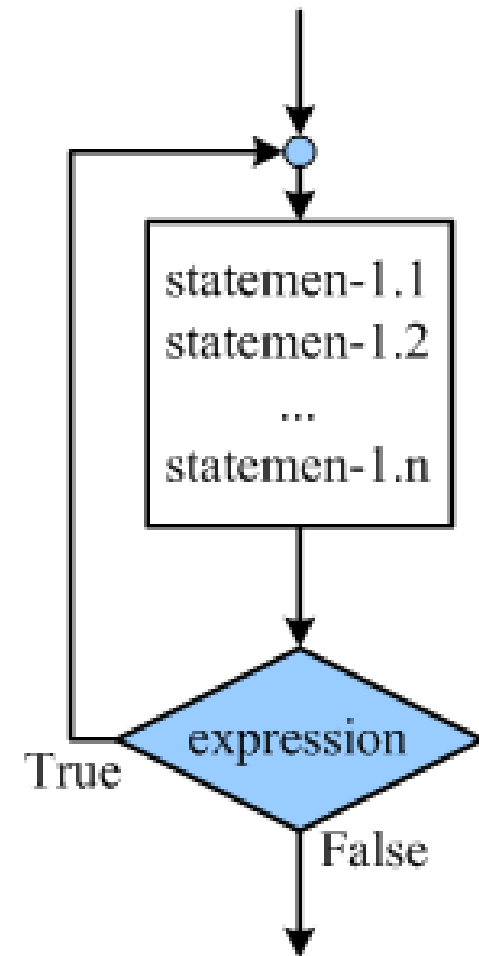
- \* **หมายเหตุ** เอกสารจากส่วนนี้เป็นต้นไป โปรแกรมตัวอย่างใดที่ เริ่มต้นด้วย int main() โดย  
หมายความว่าไม่มีพร็โพรเซสเซอร์ใดเรียกทีฟ แต่ไม่ได้เขียนแสดงไว้

## 5.3 คำสั่ง do-while

01006012 Computer Programming

```
do
    statement-1.1;
while (expression);
```

```
do
{
    statement-1.1;
    statement-1.2;
    ...
    statement-1.n;
}
while (expression);
```



## 5.3 คำสั่ง do-while

01006012 Computer Programming

เมื่อสั่งให้โปรแกรมด้านล่างทำงาน โปรแกรมจะแสดงค่าของ **i** ออกมาเท่ากับเท่าใด

```
#include <stdio.h>
int main()
{
    int i ;
    i = 0 ;
    do{
        printf("\nKMITL") ;
        i = i+1;
    } while (i > 5) ;
    printf("%d",i ) ;
    return 0 ;
}
```

→ 1

# โปรแกรม 5.4 หาผลรวม 1 ถึง 100 | do-while

01006012 Computer Programming

จงเขียนผังงานและโปรแกรมสำหรับรวมเลขจำนวนเต็ม  
ตั้งแต่ 1 – 100 โดยใช้คำสั่ง do-while

- Output Analysis
  - ผลรวมของเลขจำนวนเต็ม ตั้งแต่ 1 - 100
- Input Analysis
  - ไม่มี

# โปรแกรม 5.4 หาผลรวม 1 ถึง 100 | do-while

01006012 Computer Programming

- Process Analysis
  - โปรแกรมทำการบวกค่าเก็บไว้ในตัวแปรผลลัพธ์ แล้วเพิ่มค่าจนถึง 100
- Variable Define
  - count เป็นตัวแปรชนิดจำนวนเต็มเพื่อนับจำนวน
  - sum เป็นจำนวนเต็มเพื่อเก็บค่าผลรวม

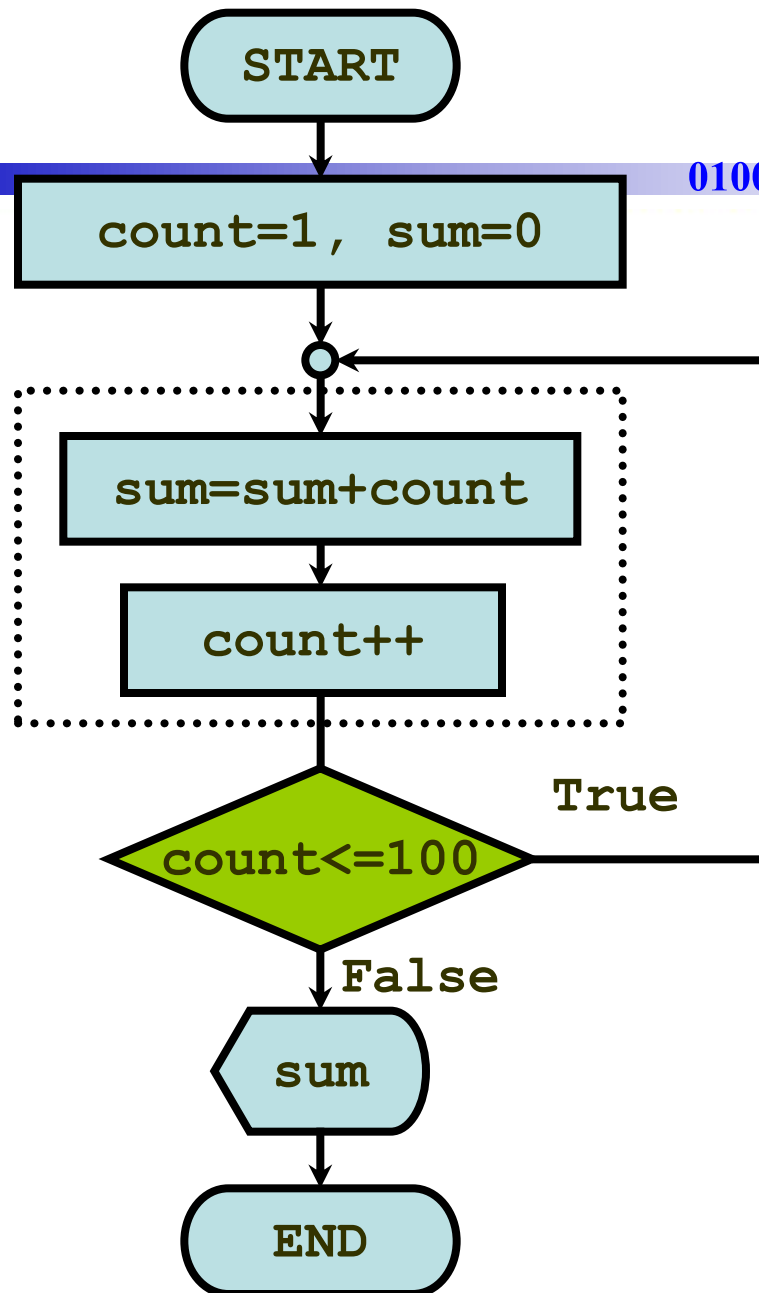
# ตัวอย่าง do-while

01006012 Computer Programming

```
int main()
{
    int    count=10;
    do
    {
        printf("%d",count) ;
        count++;
    }
    while(count<0) ;
    return 0;
}
```

Output

10





# โปรแกรม 5.4 หาผลรวม 1 ถึง 100 | do-while

01006012 Computer Programming

```
int main()
{
    int    count=1, sum=0;
    do
    {
        sum = sum + count;
        count++;
    }
    while(count<=100);
    printf ("Summation of 1 to 100 = %d", sum);
    return 0;
}
```

## 5.4 คำสั่ง for

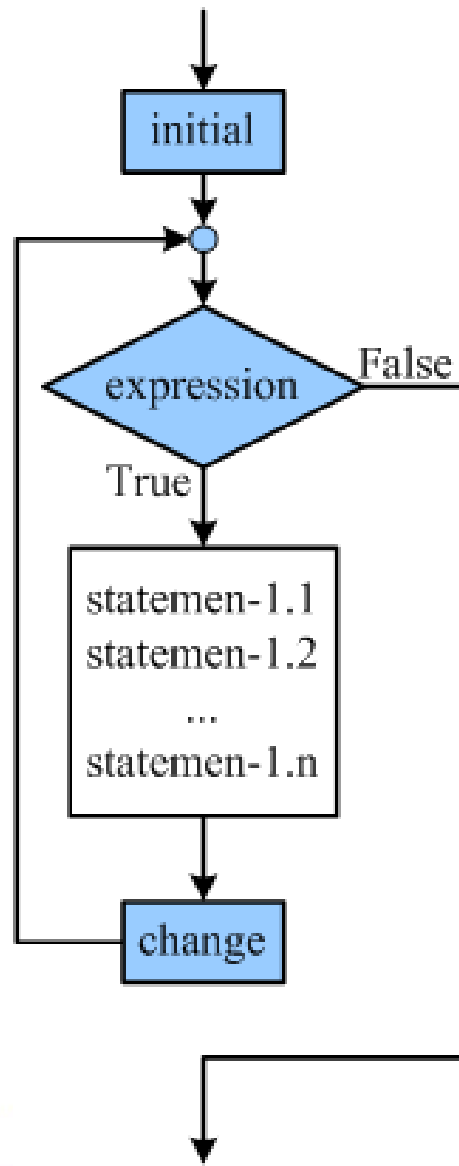
```
for (initial; expression; change)
{
    statement-1.1;
    statement-1.2;
    ...
    statement-1.n;
}
```

- *initial* เป็นส่วนที่ใช้กำหนดค่าเริ่มต้นให้กับตัวแปร
- *condition* เป็นเงื่อนไขเพื่อพิจารณา
- *change* เป็นส่วนที่เปลี่ยนแปลงค่าตัวแปร
- *statement-1, 2, ... , n* เป็นคำสั่งที่จะทำงานเมื่อเงื่อนไขเป็นจริง

# รูปแบบการใช้คำสั่ง for



01006012 Computer Programming



# โปรแกรม 5.5 หาผลรวม 1 ถึง 100 | for

01006012 Computer Programming

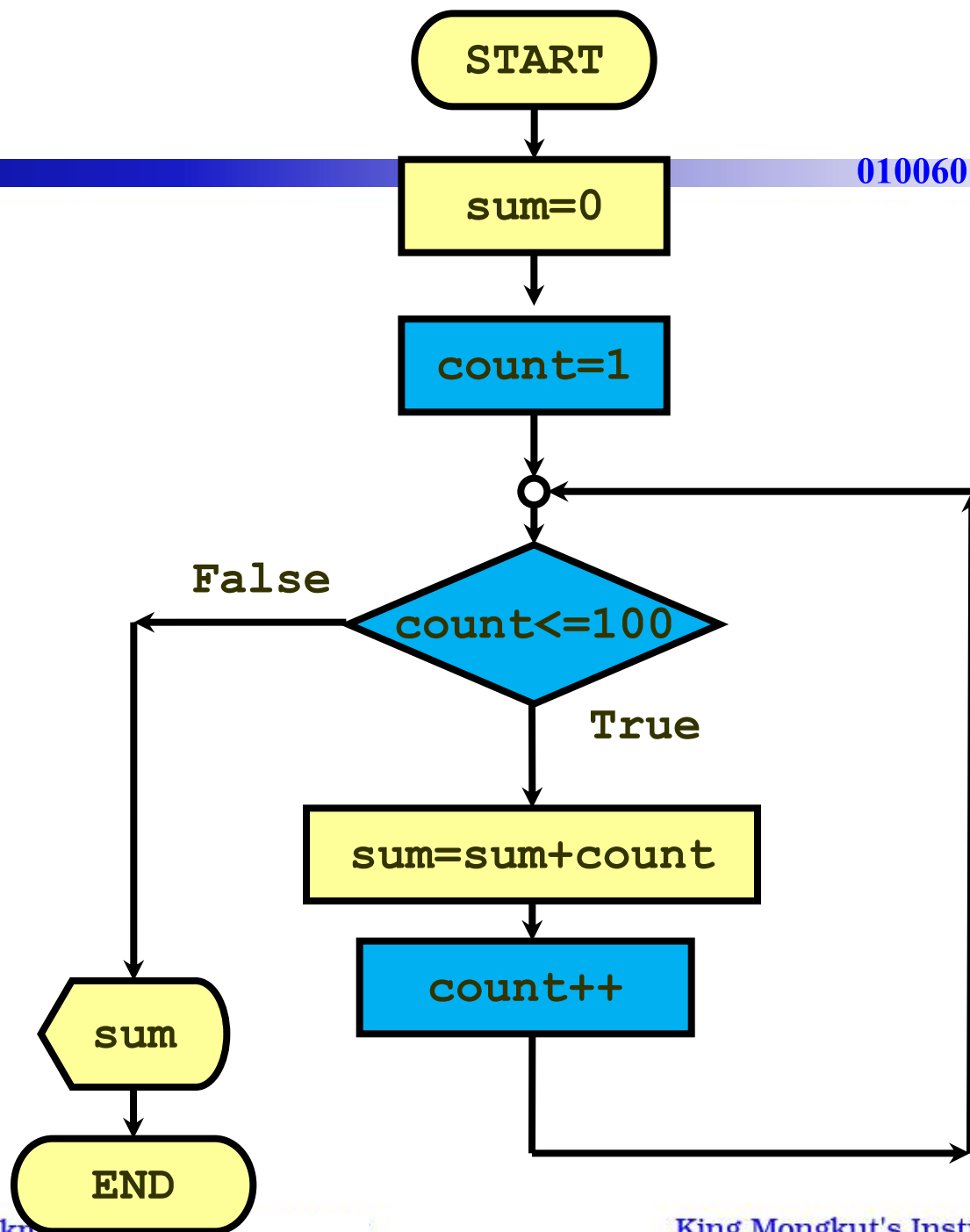
- Process Analysis

- โปรแกรมทำการบวกค่าเก็บไว้ในตัวแปรผลลัพธ์ แล้วเพิ่มค่าจนถึง 100

- Variable Define

count    เป็นตัวแปรชนิดจำนวนเต็มเพื่อนับจำนวน

sum      เป็นจำนวนเต็มเพื่อเก็บค่าผลรวม



# โปรแกรม 5.5 หาผลรวม 1 ถึง 100 | for

01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    int  sum=0, count;
    for (count=1; count<=100; count++)
    {
        sum = sum + count ;
    }
    printf ("Summation of 1 to 100 = %d",sum) ;
    return 0;
}
```

# โปรแกรม 5.6 แสดงผล a – z | for

01006012 Computer Programming

จงเขียนผังงานและโปรแกรมสำหรับแสดงผลอักษร a – z ออกทางจอภาพ โดยใช้คำสั่ง for

- Output Analysis
  - แสดงผล a – z ทางจอภาพ

```
a b c d e f g h i j k l m n o p q r s t u v w x y z
```

- Input Analysis
  - ไม่มี

# โปรแกรม 5.5 แสดงผล a – z | for

01006012 Computer Programming

- Process Analysis
  - โปรแกรมทำการวนรอบเพื่อแสดงผลอักษรตั้งแต่ a – z โดยการเพิ่มค่าตัวแปรขึ้นครั้งละ 1 (ดูตาราง ASCII Code)

- Variable Define

letter    เป็นตัวแปรชนิดอักขระ



## 5.4 คำถามท้ายบท



01006012 Computer Programming

1. จงเขียนโปรแกรมแสดงรหัสแอสกี ตั้งแต่ 33 ถึง 55

Decimal	ASCII
33	!
34	"
35	#
. . .	
. . .	
55	7

2. จงเขียนโปรแกรมรับตัวเลขเพื่อมาคำนวณหาผลบวกกำลังสอง  
จนกระทั่งตัวเลขที่รับเข้ามามีค่าเป็น 0

Enter a number : 2

Enter a number : -5

Enter a number : 0

Result : 29

## 3. ข้อใดเป็นโปรแกรมที่รันไม่รู้จบ (Infinite loop)

เมื่อกำหนด `int i=0;`

3.1 `for(i=0; i>0; i++) printf("%d",i);`

3.2 `for(i=0; i%2!=0; i += 2) puts("a");`

3.3 `while(i<7) printf("%d",i--);`

3.4 `do {  
    i+=3;  
} while(i%3==0);`

## Chapter : 5 Level : 3

Lab name : (42) to upper

### CONTENT :

จงเขียนโปรแกรมรับข้อความ แล้วแสดงผลเป็นตัวพิมพ์ใหญ่ทั้งหมดดังตัวอย่าง

ข้อความในภาษาซี จะมี character พิเศษ เพื่อแสดงถึงจุดสิ้นสุดข้อความ '\0' หรือ NULL character

ไม่ต้องแสดง NULL ออกทางจอภาพ

hint: ตัวอักษร 'a' มีค่าเป็น 97, ตัวอักษร 'A' มีค่าเป็น 65 >>>>> 97 - 65 = 32

```
1 #include <stdio.h>
2 int main() {
3     char str[300];
4     int size,i;
5     printf(" *** To Upper Case ***\n");
6     printf("Enter a string : ");
7     scanf("%[^\n]",str);
8     printf("Output : ");
9     for(size = 0; str[size] != '\0';size++){
10         for(i=0;i<size;i++) {
11             if(str[i]>=97 && str[i]<=122) {
12                 printf("%c",str[i]-32);
13             } else {
14                 printf("%c",str[i]);
15             }
16         }
17     }
18 }
```

-- output --

```
*** To Upper Case ***
Enter a string : Hello, World!
Output : HELLO, WORLD!
```

```
*** To Upper Case ***
Enter a string : wE aRe ThE loVeSiCk GiRLS
Output : WE ARE THE LOVESICK GIRLS
```

```
*** To Upper Case ***
Enter a string : The C Language is develop
```

## Chapter : 5 Level : 4

Lab name : รูปสี่เหลี่ยม ใช้แอสกี 2

ming

### CONTENT :

รับจำนวนเต็มบวก 1 จำนวนแสดงผล แล้วแสดงผลเป็นรูป สี่เหลี่ยมจัตุรัส ตามจำนวนที่รับเข้ามา

ตัวอย่าง รับเลข 3

แสดงผล 3 บรรทัด แต่ละบรรทัดแสดงผล 3 ตำแหน่ง

บรรทัดแรกเริ่มจาก ABC

บรรทัดสองเริ่มจาก BCD

บรรทัดสามเริ่มจาก CDE

```
1 #include<stdio.h>
2 int main() {
3     int r,c,n;
4     printf("Enter a number : ");
5     scanf("%d",&n);
6     for(r=0; r<n ; r++) {
7         for(c=0;c<n; c++) {
8             printf("%c",'A'+r+c);
9         }
10        printf("\n");
11    }
12    return 0;
13 }
```

-- outp

```
Enter a number : 4
ABCD
BCDE
CDEF
DEFG
```

```
Enter a number : 12
ABCDEFGHIJKL
```

## Chapter : 5 Level : 2

Lab name : แสดงผลข้อความ แบบย้อนจากหลังมาหน้า

### CONTENT :

จงเขียนโปรแกรม รับข้อความ 1 บรรทัด แล้วแสดงผล แบบย้อนกลับดังตัวอย่าง

- ข้อความในภาษาซี จะมี character พิเศษ เพื่อแสดงถึงจุดสิ้นสุดข้อความ '\0' หรือ NULL character
- การแสดงผลต้องหาตำแหน่งแรกสุดก่อน (ไม่ต้องแสดงผล NULL character)
- ตำแหน่งสุดท้าย คือ ตำแหน่ง 0

```
#include<stdio.h>
int main() {
    char str[300];
    int index;
    printf(" *** Reverse string display ***\n");
    printf("Enter a string : ");
    scanf("%[^\n]",str);
    printf("Output : ");
    for(index = 0; str[index] !='\0';index++);
    index--;
    for(index--;index>=0;index--)
        printf("%c",str[index]);

    return 0;
}
```

```
*** Reverse string display ***
Enter a string : Bangkok.
Output : .kognab
```

```
*** Reverse string display ***
Enter a string : Computer Engineering
Output : .gnireenignE retupmoC
```

```
*** Reverse string display ***
Enter a string : The quick brown fox
```