

Computer Programming II
การเขียนโปรแกรมคอมพิวเตอร์ 2
LECTURE#5 คำสั่งควบคุมการวนซ้ำ (Loop Statement)

อ.สถิตย์ ประสมพันธ์
ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ
KMUTNB

คำสั่งควบคุม (Control Statement)

- การวนซ้ำ หมายถึง การควบคุมให้การกระทำบางอย่างคำสั่งซ้ำหลายรอบ ซึ่งจะช่วยให้การเขียนโปรแกรมทำได้ง่ายสะดวก ไม่ต้องเขียนคำสั่งเดิมหลายครั้ง ทำให้โปรแกรมมีความกระชับ สามารถตรวจสอบความผิดพลาดได้ง่าย
- คำสั่งการวนรอบมี 3 ชนิดหลัก ๆ คือ
 - คำสั่ง for
 - คำสั่ง do-while
 - คำสั่ง while

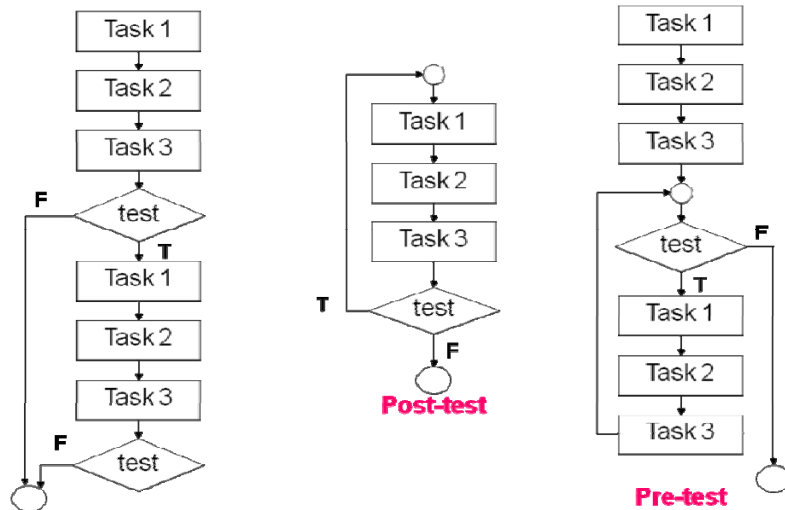
คำสั่งควบคุม (Control Statement)

- แต่ละคำสั่งมีรูปแบบและวิธีการใช้งานที่แตกต่างกัน สามารถเลือกใช้ตามความเหมาะสมของการใช้งานในโปรแกรม
- ส่วนประกอบของคำสั่งแบบวนซ้ำ ประกอบด้วย
 - ส่วนของการตรวจสอบ (loop test) เป็นเงื่อนไขเพื่อทดสอบว่าจะทำวนซ้ำอีกหรือไม่
 - ส่วนของการทำวนซ้ำ (loop body) เป็นชุดคำสั่งที่จะถูกดำเนินการ

รายละเอียดของการสร้าง loop

- รายละเอียดของการสร้าง loop มีขั้นตอนการทำงานดังนี้
 - ระบุส่วนของการทำงานที่ต้องทำซ้ำ (loop body) ในส่วนนี้จะเป็นการระบุเงื่อนไข (loop test) ที่จะทำซ้ำ หรือเลิกทำซ้ำ
 - ระบุชนิดของ loop ที่จะใช้ ซึ่งชนิดของการทำซ้ำจะมีประเภทที่ต่างกันหลัก ๆ 2 ประเภทคือ
 - Pre-test loop
 - Post-test loop

รายละเอียดของการสร้าง loop



คำสั่ง for

คำสั่ง for เป็นคำสั่งที่สั่งให้ทำคำสั่ง หรือกลุ่มของคำสั่งวนซ้ำหลายรอบ โดยมีจำนวนรอบในการวนซ้ำที่แน่นอน

รูปแบบ

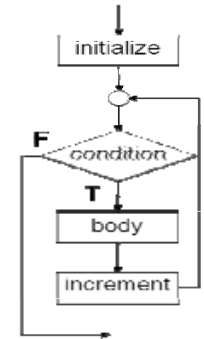
for (ค่าเริ่มต้น; เงื่อนไข; ค่าเพิ่มหรือค่าลด)

{

คำสั่ง

}

โดยที่ ค่าเริ่มต้น, เงื่อนไข, ค่าเพิ่มหรือค่าลด เป็นนิพจน์ คำสั่ง หมายถึง คำสั่งที่จะถูกกระทำซ้ำ ซึ่งอาจจะมีเพียงคำสั่งเดียว หรือหลายคำสั่งก็ได้

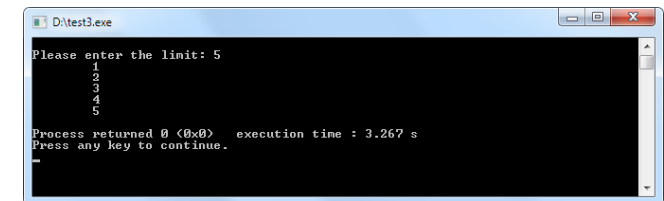


รูปแบบการใช้ for แบบต่างๆ

```
#include<stdio.h>
int main()
{
    int i=0;
    float sum=0, score=0, sc=0;
    // ตัวอย่างการใช้คำสั่ง for แบบที่ 1
    for (i=1, sum=0; i<=10; i++)
        sum += i;
    // ตัวอย่างการใช้คำสั่ง for แบบที่ 2
    for (i=1; i<=25; i++)
    {
        scanf("%f", &score);
        printf("%f\n", score);
    }
    // ตัวอย่างการใช้คำสั่ง for แบบที่ 3
    for (scanf("%f", &sc), sum=0;
        sc>=0; scanf("%f", &sc))
        sum += sc;
    return 0;
}
```

ตัวอย่างการพิมพ์ตัวเลข 1 ถึง จำนวนที่ผู้ใช้ระบุ

```
#include <stdio.h>
int main (void)
{
    // Local Declarations
    int limit;
    int i;
    // Statements
    printf ("\nPlease enter the limit: ");
    scanf ("%d", &limit);
    for (i = 1; i <= limit; i++)
    {
        printf ("%t%d\n", i);
    }
    return 0;
} // main
```



การวนซ้ำเพื่อพิมพ์ตัวเลขในแต่ละบรรทัด

```
#include <stdio.h>
int main (void)
{
    // Statements
    for (int i = 1; i <= 3; i++)
    {
        printf("Row %d: ", i);
        for (int j = 1; j <= 5; j++)
            printf("%3d", j);
        printf("\n");
    } // for i
    return 0;
} // main
```

ผลลัพธ์จากการประมวลผลโปรแกรม

```
Row 1:  1  2  3  4  5
Row 2:  1  2  3  4  5
Row 3:  1  2  3  4  5
```

คำสั่ง while

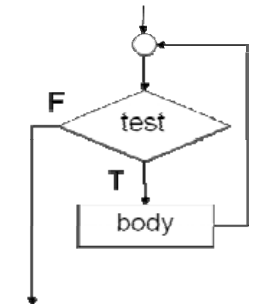
คำสั่ง while เป็นคำสั่งวนซ้ำ ที่สั่งให้ทำคำสั่งที่อยู่ภายในคำสั่ง while หลายรอบจนกระทั่งเงื่อนไขเป็นเท็จ หรือ 0 จึงจะจบการวนซ้ำ

รูปแบบ

while (เงื่อนไข)

```
{
    คำสั่ง
}
```

โดยที่ เงื่อนไข เป็นนิพจน์ที่ให้ผลลัพธ์เป็นจริงหรือเท็จ
คำสั่งอยู่ภายในคำสั่ง while อาจมีเพียงคำสั่งเดียว หรือหลายคำสั่ง



ตัวอย่างของการใช้คำสั่ง while ในรูปแบบต่าง ๆ

```
#include <stdio.h>
int main (void)
{
    int x;
    float sum=0;
    float vat=0;
    int prc=0;
    scanf ("%d",&x);
    while (x<0)
        x++;
    while (sum<20)
    {
        scanf ("%d", &prc);
        if (prc<20)
        {
            vat = .07*prc;
            sum += prc + vat;
        }
    }
    return 0;
} // main
```

ตัวอย่างการพิมพ์ตัวเลขที่ผู้ใช้ระบุ โดยพิมพ์จากตัวนั้น ๆ ถึง 1 บรรทัดละ 10 ตัวเลข

```
#include <stdio.h>
int main (void)
{
    // Local Declarations
    int num;
    int lineCount;
    // Statements
    printf ("Enter an integer
between 1 and 100: ");
    scanf ("%d", &num);
    // Initialization
    // Test number
    if (num > 100)
        num = 100;
    lineCount = 0;
    while (num > 0)
    {
        if (lineCount < 10)
            lineCount++;
        else
        {
            printf("\n");
            lineCount = 1;
        } // else
        printf("%4d", num--);
    } // while
    return 0;
} // main
```

ผลลัพธ์จากการประมวลผลโปรแกรม

```
Enter an integer between 1 and 100: 15
15 14 13 12 11 10  9  8  7  6
 5  4  3  2  1
```

ตัวอย่างการรับค่าตัวเลขและแสดงผลตัวเลข หาจำนวน digits

และผลรวมของทุก ๆ digits

```
#include <stdio.h>
int main (void)
{
    int number;
    int count = 0;
    int sum = 0;
    printf("Enter an integer: ");
    scanf ("%d", &number);
    printf("Your number is:  %d\n\n", number);
    while (number != 0)
    {
        count++;
        sum += number % 10; //->5
        number /= 10; //->1234
    } // while
    printf("The number of digits is: %3d\n", count);
    printf("The sum of the digits is: %3d\n", sum);
    return 0;
} // main
```

ผลลัพธ์จากการประมวลผลโปรแกรม

```
Enter an integer: 12345
Your number is: 12345

The number of digits is: 5
The sum of the digits is: 15
```

ตัวอย่างการรับค่าตัวเลขแล้วพิมพ์ตัวเลขนั้น ๆ จากหลัมาหน้า

```
#include <stdio.h>
int main (void)
{
    // Local Declarations
    long num;
    int digit;
    // Statements
    printf("Enter a number and I'll print it backward: ");
    scanf ("%d", &num);

    while (num > 0)
    {
        digit = num % 10;
        printf("%d", digit);
        num = num / 10;
    } // while
    printf("\nHave a good day.\n");
    return 0;
} // main
```

คำสั่ง do-while

คำสั่ง do-while เป็นคำสั่งวนซ้ำ ที่สั่งให้ทำคำสั่งที่อยู่ภายในคำสั่ง do-while หนึ่งรอบ แล้วจึงจะตรวจสอบเงื่อนไข ถ้าเงื่อนไขเป็นเท็จจะจบการทำงานทันที

รูปแบบ

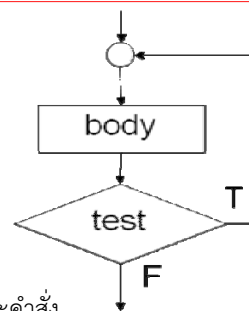
do

{

คำสั่ง;

} while (เงื่อนไข);

โดยที่ เงื่อนไข เป็นนิพจน์ที่ให้ผลลัพธ์เป็นจริงหรือเท็จ และคำสั่ง
อยู่ภายในคำสั่ง do-while อาจมีเพียงคำสั่งเดียว หรือหลายคำสั่ง



ตัวอย่างการใช้คำสั่ง while และ do-while เพื่อพิมพ์ตัวเลขจาก 5 ถึง 1

```
#include <stdio.h>
int main (void)
{
    // Local Declarations
    int loopCount;
    // Statements
    loopCount = 5;
    printf("while loop : ");
    while (loopCount > 0)
        printf ("%3d", loopCount--);
    printf("\n\n");
    loopCount = 5;
    printf("do...while loop: ");
    do
        printf ("%3d", loopCount--);
    while (loopCount > 0);
    printf("\n");
    return 0;
} // main
```

ผลลัพธ์จากการประมวลผลโปรแกรม

```
while loop : 5 4 3 2 1
do...while loop: 5 4 3 2 1
```

การวนซ้ำซ้อน

การวนซ้ำซ้อน หมายถึง การควบคุมให้กระทำบางคำสั่งหลายรอบ และในการทำงานแต่ละรอบก็จะควบคุมให้ทำคำสั่งที่อยู่ภายในนั้นอีกหลายรอบ การวนซ้ำซ้อน อาจจะเขียนได้โดยการใช้คำสั่งวนซ้ำใด ๆ ซ้อนกัน 2 คำสั่ง เช่น

คำสั่ง for ซ้อนกับคำสั่ง for

คำสั่ง while ซ้อนกับคำสั่ง while

คำสั่ง while-do ซ้อนกับคำสั่ง while-do

การวนซ้ำซ้อน

```
#include <stdio.h>
int main (void)
{
    // Local Declarations
    int limit;
    // Read limit
    printf("\nPlease enter a number between 1 and 9: ");
    scanf("%d", &limit);
    for (int lineCtrl = 1; lineCtrl <= limit; lineCtrl++)
    {
        for (int numCtrl = 1;
            numCtrl <= lineCtrl;
            numCtrl++)
            printf("%1d", numCtrl);

        printf("\n");
    } // for lineCtrl
    return 0;
} // main
```

ผลลัพธ์จากการประมวลผลโปรแกรม

Please enter a number between 1 and 9: 6
1
12
123
1234
12345
123456