

## การทดลองที่ 5

### การเขียนโปรแกรมแบบวนซ้ำ

#### วัตถุประสงค์

1. เข้าใจกระบวนการทำงานแบบวนซ้ำ
2. สามารถเขียนโปรแกรมเพื่อให้เกิดการทำงานแบบวนซ้ำได้
3. สามารถประยุกต์ใช้การทำงานแบบวนซ้ำเพื่อแก้ปัญหาโจทย์ได้

#### ทฤษฎีโดยย่อ

##### ก. คำสั่งวนรอบในภาษาซี

คำสั่งที่ทำให้เกิดการวนรอบในภาษามีอยู่ 3 คำสั่ง ได้แก่ คำสั่ง for, while และ do...while โดยคำสั่ง for กับ while นั้นมีลักษณะการทำงานคล้ายกัน ดังนั้น 2 คำสั่งนี้จึงแทบจะใช้แทนกันได้ทุกกรณี โดยต้องแก้ไขชุดคำสั่งที่อยู่ในการวนรอบนั้นเพียงเล็กน้อย ส่วนคำสั่ง do...while นั้นจะมีลักษณะการทำงานที่แตกต่างออกไป ดังนั้นการใช้ do...while แทนคำสั่ง while หรือ for จะต้องมีการปรับแก้คำสั่งต่าง ๆ พอสมควร

##### ข. รูปแบบการใช้คำสั่งวนรอบ

คำสั่ง while คำสั่งนี้จะมีพารามิเตอร์เดียวซึ่งเป็นเงื่อนไขที่จะทำให้เกิดการวนรอบหรือหยุดการวนรอบ พารามิเตอร์นี้จะอยู่ในรูปของการเปรียบเทียบทางตรรกศาสตร์ ดังนี้

```
while (เงื่อนไข)
```

```
    คำสั่งในภาษาซี;
```

คำสั่ง while จะวนรอบไปเรื่อย ๆ ตราบเท่าที่เงื่อนไขให้ค่าทางตรรกศาสตร์เป็น จริง (True) และการวนรอบของคำสั่ง while จะหยุดเมื่อเงื่อนไขให้ค่าเป็น เท็จ (False) คำสั่ง while นี้จะทำการตรวจสอบเงื่อนไขก่อนที่จะเริ่มทำงานในรอบใหม่

คำสั่ง for คำสั่งนี้พารามิเตอร์ของคำสั่งจะประกอบไปด้วย 3 ส่วนซึ่งอยู่ในวงเล็บเดียวกัน โดยแต่ละส่วนจะกันด้วยเครื่องหมายอัฒภาค (semi colon ;)

```
for (การกำหนดค่าเริ่มต้นของ counter; เงื่อนไข; การเพิ่มค่า counter)
    คำสั่งในภาษาซี;
```

เงื่อนไขของคำสั่ง for มีหน้าที่ควบคุมให้เกิดการวนรอบหรือหยุดการวนรอบ โดยลักษณะของเงื่อนไขจะเหมือนกับของคำสั่ง while และการตรวจสอบเงื่อนไขจะกระทำก่อนที่จะเริ่มทำงานในรอบใหม่เช่นเดียวกับคำสั่ง while โดยมากมักจะใช้ค่าจากตัวแปรที่ทำหน้าที่นับรอบ (counter) มาเป็นส่วนหนึ่งของเงื่อนไข ส่วนตัวแปรที่ทำหน้าที่นับรอบ (counter) จะถูกกำหนดค่าเริ่มต้นที่พารามิเตอร์ตัวแรกของ for และจะถูกเปลี่ยนแปลงค่าโดยพารามิเตอร์ตัวที่ 3 ของคำสั่ง for

คำสั่ง do...while เป็นคำสั่งที่ทำให้เกิดการวนรอบเช่นเดียวกับ 2 คำสั่งข้างต้น โดยมีส่วนที่เหมือนกับคำสั่ง for และ while คือมีพารามิเตอร์ที่เป็นเงื่อนไขการวนรอบอยู่ในรูปของการเปรียบเทียบทางตรรกศาสตร์ หากแต่การตรวจสอบเงื่อนไขของคำสั่ง do...while จะกระทำหลังจากที่จบการทำงานแต่ละรอบ โดยถ้าเงื่อนไขให้ค่าเป็นจริง การวนรอบจะดำเนินต่อไป หากเงื่อนไขให้ค่าเป็นเท็จ การวนรอบก็จะยุติลง รูปแบบการใช้งานคำสั่ง do...while เป็นดังนี้

```
do
{
    คำสั่งในภาษาซี;
} while (เงื่อนไข);
```

## ตอนที่ 1 ศึกษาการวนรอบ

### 1.1 ทำความเข้าใจเกี่ยวกับพื้นฐานของการวนรอบ

- 1) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int x;
    x=1;                                //00
    while (x<5)                          //01
    {
        printf("Loop Test\n");
        x++;                             //02
    }
    printf("\nAfter loop end, x=%d",x);  //03
    return 0;
}
```

- ก) นศ. คาดว่าจะมีการแสดงผลประโยค Loop Test ที่กี่ครั้ง \_\_\_\_\_
- ข) ให้ นศ. Compile และ run โปรแกรมนี้ เพื่อตรวจสอบคำตอบในข้อ ก)
- ค) ให้ นศ. เปลี่ยนคำสั่ง while ในบรรทัด //01 เป็น while (x<=5) นศ. คาดว่าโปรแกรมจะแสดงข้อความ Loop Test จำนวนเท่ากับที่แสดงในข้อ ข) หรือไม่เพราะเหตุใด \_\_\_\_\_
- ง) ให้ นศ. Compile และ run โปรแกรมที่แก้ตามข้อ ค) เพื่อตรวจสอบคำตอบของข้อ ค)
- จ) หากเปลี่ยนบรรทัด //00 เป็น x=0; จะได้ข้อความ Loop Test ที่กี่ครั้ง \_\_\_\_\_
- ฉ) หากลบบรรทัด //00 จะทำให้ \_\_\_\_\_
- เพราะ \_\_\_\_\_
- ช) จากโปรแกรมที่สมบูรณ์ (มีครบทั้งบรรทัด //00 //01 //02 และ //03) หากลบบรรทัด //02 จะทำให้ \_\_\_\_\_
- เพราะ \_\_\_\_\_

ข) ในแต่ละรอบที่มีการแสดงข้อความ Loop Test ค่าของ x มีการเปลี่ยนแปลงหรือไม่  
อย่างไร \_\_\_\_\_

ฅ) หากต้องการให้โปรแกรมนี้แสดงข้อความ Loop Test 10 ครั้งจะต้องแก้ไขโปรแกรม  
อย่างไร \_\_\_\_\_

ญ) ให้ นศ. สรุปผลของตัวแปร x ที่มีต่อโปรแกรมนี้ \_\_\_\_\_

2) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>

int main()
{
    int x=1;
    while (x<10)                                //00
    {
        printf("x=%d\n",x);
        x++;                                    //01
    }
    printf("After Loop end, x=%d",x); //02
    return 0;
}
```

ก) นศ. คาดว่าการวนรอบของโปรแกรมนี้เกิดขึ้นกี่รอบ \_\_\_\_\_

เมื่อ นศ. Compile & Run จะได้ว่าโปรแกรมนี้ทำการวนรอบกี่รอบ \_\_\_\_\_

ข) ค่าของ x ที่แสดงบนหน้าจอโดย printf ในบรรทัด //02 \_\_\_\_\_

ให้ นศ. เปรียบเทียบค่า x ที่แสดงโดยบรรทัด //02 กับเงื่อนไขในบรรทัด //00 \_\_\_\_\_

ค) จากข้อ ข) นศ. เห็นสาเหตุที่ทำให้การวนรอบยุติหรือไม่ อย่างไร \_\_\_\_\_

ง) เปลี่ยนบรรทัด //01 เป็น  $x+=2$ ; เมื่อสั่งให้โปรแกรมทำงานแล้วจะได้ผลลัพธ์บนหน้าจอที่บรรทัด ต่างจากผลของข้อ ก) และ ข) อย่างไร \_\_\_\_\_

จ) เปลี่ยนบรรทัด //01 เป็น  $x+=4$ ; เมื่อสั่งให้โปรแกรมทำงานแล้วจะได้ผลลัพธ์แตกต่างจากข้อ ก) ข) และ ง) อย่างไร \_\_\_\_\_

ฉ) สรุปผลของการเปลี่ยนแปลงค่า  $x$  ที่มีต่อการทำงานของโปรแกรมนี้อย่างไร \_\_\_\_\_

ช) จำนวนการวนรอบของคำสั่ง while ขึ้นอยู่กับค่า  $x$  หรือไม่ อย่างไร \_\_\_\_\_

ซ) หากลบบรรทัด //01 จะเกิดผลอย่างไร \_\_\_\_\_

ฌ) จากข้อ ซ) บรรทัด //01 มีความสำคัญอย่างไรต่อโปรแกรมและต่อคำสั่ง while \_\_\_\_\_

ฎ) หากแก้ไขบรรทัด //00 เป็น while ( $x=10$ ) เมื่อโปรแกรมทำงานจะแสดงผลออกมาที่บรรทัด \_\_\_\_\_

ฏ) จากข้อ ฎ) เพราะเหตุใด \_\_\_\_\_

3) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int x;
    x = ____;           //01
    while (x < 100)      //02
    {
        printf("Hello there\n");
        x++;            //03
    }
    return 0;
}
```

ก) หากต้องการให้โปรแกรมแสดงข้อความ Hello there บนหน้าจอ 7 ครั้ง ในบรรทัด

//01 ต้องกำหนดค่า x เป็นเท่าใด \_\_\_\_\_

ข) ให้บอกวิธีหาค่าตัวแปร x เมื่อคำสั่ง while ทำงานจบ \_\_\_\_\_

ค) เมื่อคำสั่ง while จบการทำงาน ค่าตัวแปร x มีค่าเท่าใด \_\_\_\_\_

ง) หากกำหนดค่าในบรรทัด //01 เป็น x=0; ให้แก้ไขการทำงานของคำสั่ง while

ในบรรทัด //02 เพื่อให้โปรแกรมแสดงข้อความ Hello there บนหน้าจอ 9 ครั้ง \_\_\_\_\_

จ) จากข้อ ง) เมื่อคำสั่ง while ทำงานเสร็จ ตัวแปร x มีค่าเท่าใด \_\_\_\_\_

ฉ) หากเปลี่ยนบรรทัด //03 เป็น x+=6; ให้แก้ไขบรรทัด //01 และ //02 เพื่อให้

โปรแกรมแสดงข้อความ Hello there บนหน้าจอ 6 ครั้ง \_\_\_\_\_

ช) จากข้อ ฉ) เมื่อคำสั่ง while ทำงานเสร็จ ตัวแปร x มีค่าเท่าใด \_\_\_\_\_

ซ) หากต้องการให้โปรแกรมแสดงข้อความ Hello there บนหน้าจอ 6 ครั้ง และเมื่อจบ

การทำงานของคำสั่ง while ตัวแปร x มีค่าเป็น 83 ต้องแก้บรรทัด //01 และ //02 เป็น

4) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int i=0,x;
    printf("Input number: ");
    scanf("%d",&x);
    while (i<x)
    {
        printf("Loop: %d\n", i);
        i++;
    }
    return 0;
}
```

ก) ตัวแปรที่ทำหน้าที่นับจำนวนรอบการทำงาน (Loop counter) ของคำสั่ง while คือตัว

แปรใด \_\_\_\_\_

ข) เมื่อ run โปรแกรมแล้วป้อนค่า 10 คำสั่ง while จะวนรอบกี่ครั้ง ทราบได้อย่างไร \_\_\_\_

\_\_\_\_\_

ค) หาก run โปรแกรม แล้วป้อนค่า 7 คำสั่ง while จะวนรอบกี่ครั้ง \_\_\_\_\_

ง) หาก run โปรแกรม แล้วป้อนค่า 22 คำสั่ง while จะวนรอบกี่ครั้ง \_\_\_\_\_

จ) จากข้อ ข) – ง) ค่าที่ป้อนให้กับโปรแกรมมีผลต่อการวนรอบของคำสั่ง while หรือไม่

เพราะเหตุใด \_\_\_\_\_

\_\_\_\_\_

ฉ) หากลบบรรทัด //01 จะทำให้ \_\_\_\_\_

เพราะ \_\_\_\_\_

ช) หากเปลี่ยนบรรทัด //01 เป็น i+=9; เมื่อ run โปรแกรมแล้วต้องป้อนค่าใด

โปรแกรมจึงจะแสดงผล 8 บรรทัด \_\_\_\_\_

- 5) หากต้องการให้ได้ผลการทำงานดังในรูปด้านล่างจะต้องแก้ไขโปรแกรมในข้อ 4 อย่างไร

```
C:\ Select C:\WINDOWS\system32\cmd.exe
Loop: 9
Loop: 15
Loop: 21
Loop: 27
Loop: 33
Loop: 39
Loop: 45
Loop: 51
Loop: 57
Loop: 63
Loop: 69
Loop: 75
Loop: 81
Loop: 87
Loop: 93
Loop: 99
Loop: 105
Press any key to continue . . .
```

- 6) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int i,x;
    printf("Input number: ");
    scanf("%d", &x);
    for(i=0;i<x;i++)
        printf("Loop: %d\n", i);
    return 0;
}
```



ก) โปรแกรมนี้กับโปรแกรมในข้อ 4) ให้ผลการทำงานเหมือนกันหรือไม่ \_\_\_\_\_

ข) ตัวแปรที่ใช้ในรอบการทำงานของคำสั่ง for คือ \_\_\_\_\_

ค) ตัวแปร i ถูกกำหนดค่าเมื่อใด \_\_\_\_\_

ง) หากแก้ไขบรรทัด //01 เป็น `for(i=0;i=108;i++)` เมื่อ run โปรแกรมแล้ว  
ป้อนค่า 10 โปรแกรมจะแสดงผลบนหน้าจอที่บรรทัด \_\_\_\_\_

จ) หากแก้ไขบรรทัด //01 เป็น `for(i=0;i<x;i+2)` เมื่อ run โปรแกรมแล้วป้อน  
ค่า 10 โปรแกรมจะแสดงผลบนหน้าจอที่บรรทัด \_\_\_\_\_

ฉ) หากแก้ไขบรรทัด //01 เป็น `for(i=0;i<x;i==2)` เมื่อ run โปรแกรมแล้ว  
ป้อนค่า 10 โปรแกรมจะแสดงผลบนหน้าจอที่บรรทัด \_\_\_\_\_

ช) จากข้อ ง) - ฉ) เหตุใดจึงเป็นเช่นนั้น \_\_\_\_\_

---

---

---

ซ) หากแก้ไขบรรทัด //01 เป็น `for(i=10;i<x;i++)` เมื่อ run โปรแกรมแล้ว  
ป้อนค่า 5 โปรแกรมจะแสดงผลบนหน้าจอที่บรรทัด \_\_\_\_\_

ฌ) จากข้อ ซ) หาก run โปรแกรมแล้วป้อน 15 โปรแกรมจะแสดงผลบนหน้าจอที่  
บรรทัด \_\_\_\_\_

ญ) จากข้อ ซ) - ฌ) เหตุใดจึงเป็นเช่นนั้น \_\_\_\_\_

---

---

---

7) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int x=0,y;                                //01
    printf("Input number: ");
    scanf("%d", &y);
    do
    {
        printf("Loop: %d\n", x);
        x++;
    } while(x<y);
    return 0;
}
```

ก) โปรแกรมนี้กับโปรแกรมในข้อ 4) ให้ผลการทำงานเหมือนกันหรือไม่ \_\_\_\_\_

ข) โปรแกรมนี้กับโปรแกรมในข้อ 6) ให้ผลการทำงานเหมือนกันหรือไม่ \_\_\_\_\_

ค) ตัวแปรที่ใช้ในรอบการทำงานของคำสั่ง do...while คือ \_\_\_\_\_

ง) เมื่อ run โปรแกรมแล้วป้อนค่า 5 โปรแกรมจะแสดงผลกี่บรรทัด \_\_\_\_\_

จ) หากแก้ไขบรรทัด //01 เป็น int x=10, y; เมื่อ run โปรแกรมแล้วป้อนค่า 5  
โปรแกรมจะแสดงผลกี่บรรทัด เพราะเหตุใด \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

8) ให้สรุปสาเหตุของการเกิดการวนรอบไม่รู้จบ โดยอาศัยผลการทดลองที่ได้ทำมาแล้วข้างต้น

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

## 1.2 การใช้คำสั่งวนรอบเพื่อทำงานอย่างง่าย ๆ

- 1) ใช้โปรแกรมต่อไปนี้ตอบคำถามด้านล่าง

```
#include<stdio.h>
int main()
{
    int i, x, sum=0;
    printf("Input number: ");
    scanf("%d", &x);
    for(i=1;i<=x;i++)
    {
        sum+=i;
        printf("Loop:%d, sum=%d\n",i,sum);    //01
    }
    printf("Summation from 1 to %d = %d", x, sum);
    return 0;
}
```

- ก) เมื่อ run โปรแกรมแล้วป้อนค่า 3 จะได้ผลลัพธ์บนหน้าจอคือ \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

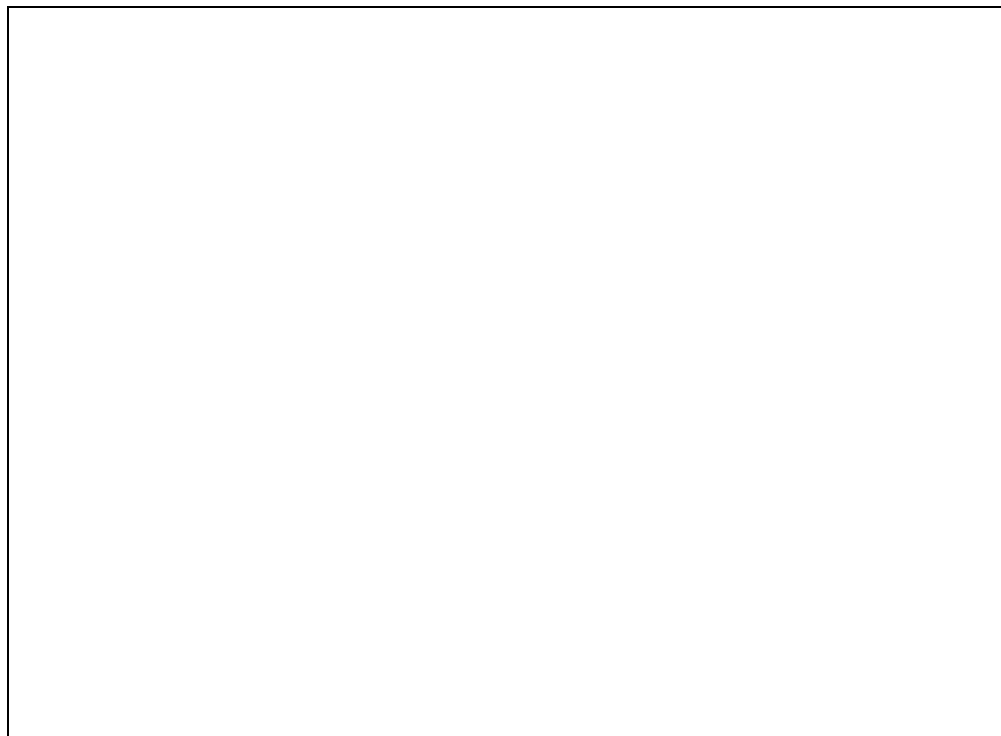
- ข) ให้เขียนโปรแกรมนี้ใหม่โดยเปลี่ยนไปใช้คำสั่ง while แทนการใช้คำสั่ง for

ค) ให้เขียนโปรแกรมนี้ใหม่โดยเปลี่ยนไปใช้คำสั่ง do...while แทนการใช้คำสั่ง for



2) ให้ดัดแปลงแก้ไขโปรแกรมในข้อ 1) ให้เป็นโปรแกรมหาค่าเฉลี่ยของตัวเลขตั้งแต่ 1 ถึงค่าที่

ป้อนให้กับโปรแกรม



## ตอนที่ 2 การฝึกประยุกต์ใช้งานคำสั่งวนรอบ

- 1) เขียนโปรแกรมรับเลขจำนวนเต็ม 1 ค่า แล้วให้แสดงผลดังตัวอย่างด้านล่าง (ให้ฝึกใช้คำสั่งวนรอบทั้ง 3 คำสั่ง)

```
Select C:\WINDOWS\system32\cmd.exe
Input number: 12
1 2 3 4 5 6 7 8 9 10 11 12
Press any key to continue . . .
```

ใช้คำสั่ง for

ใช้คำสั่ง while

ใช้คำสั่ง do...while

- 2) เขียนโปรแกรมให้รับค่า 1 ค่า แล้วแสดงแม่สูตรคูณตามค่าที่รับเข้ามานั้น (ให้ฝึกใช้งานคำสั่งวนรอบทั้ง 3 คำสั่ง)

```
CA: Select C:\WINDOWS\system32\cmd.exe
Input number: 12
12*1=12
12*2=24
12*3=36
12*4=48
12*5=60
12*6=72
12*7=84
12*8=96
12*9=108
12*10=120
12*11=132
12*12=144
Press any key to continue . . .
```

|                      |
|----------------------|
| ใช้คำสั่ง for        |
| ใช้คำสั่ง while      |
| ใช้คำสั่ง do...while |

- 3) ให้เขียนโปรแกรมรับค่าตัวเลข โดยโปรแกรมจะรับค่าตัวเลขไปจนกระทั่งป้อนค่า 0 ให้กับ

โปรแกรม โปรแกรมจึงจะหยุดรับค่า แล้วแสดงผลรวมของตัวเลขทั้งหมดที่รับเข้ามา

C:\ Select C:\WINDOWS\system32\cmd.exe

```
Input number: 1
Input number: 2
Input number: 3
Input number: -4
Input number: 5
Input number: -6
Input number: 7
Input number: 8
Input number: 9
Input number: 0
Total=25
Press any key to continue . . .
```