

Repetition Structure

ฉัตรชัย เกษมทวีโชค

Chatchai.kase@ku.th

โครงสร้างวนซ้ำ (Repetition Structure)

for...loop

- กำหนดรอบด้วยตัวแปรจำนวนนับ
- ควบคุมรอบด้วยการเพิ่มค่าหรือลดค่าตัวแปรจำนวนนับ
- เงื่อนไขในการหยุดคือ เมื่อค่าตัวแปรมีค่าครบตามเงื่อนไขการควบคุม

while Loop

- ตรวจสอบเงื่อนไขเป็นจริงก่อนวนการทำงาน
- จำนวนนับไม่แน่นอน
- ใช้เงื่อนไขทางตรรกศาสตร์ในการควบคุมรอบและการหยุดการทำงาน

do...while

- วนการทำงานก่อนตรวจสอบเงื่อนไขเป็นจริง
- จำนวนนับไม่แน่นอน
- ใช้เงื่อนไขทางตรรกศาสตร์ในการควบคุมรอบและการหยุดการทำงาน

for..loop

- กำหนดรอบด้วยตัวแปรจำนวนนับ หรือสมาชิกในกลุ่ม

```
for val in sequence :
```

```
    statement1  
    statement2  
    statement3  
    ...
```

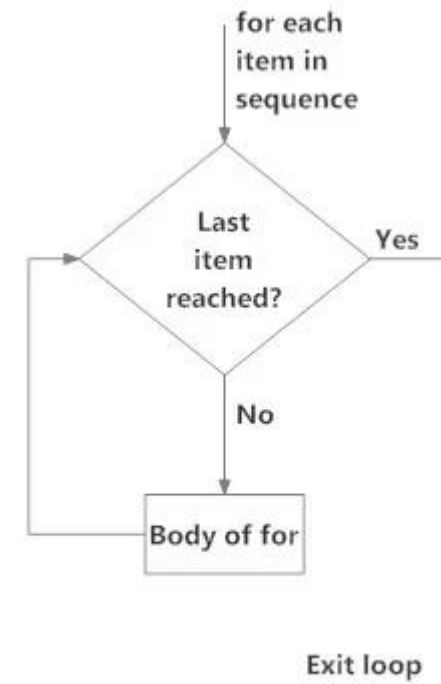


Fig: operation of for loop

for..loop

- กำหนดรอบด้วยตัวแปรจำนวนนับ หรือสมาชิกในกลุ่ม

```
# Program to find the sum of all numbers stored in a list

# List of numbers
numbers = [6, 5, 3, 8, 4, 2, 5, 4, 11]

# variable to store the sum
sum = 0

# iterate over the list
for val in numbers:
    sum = sum + val

print("The sum is", sum)
```

Source: <https://www.programiz.com/python-programming/for-loop>

Python For Loops

คำสั่ง **continue** ใช้หยุดรอบปัจจุบันแล้วเริ่มวนไปรอบถัดไปของคำสั่ง **for**

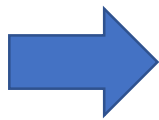
```
fruits = ["apple", "banana", "cherry"]  
for x in fruits:  
    if x == "banana":  
        continue  
    print(x)
```



apple
cherry

คำสั่ง **break** ใช้หยุดการวนซ้ำของคำสั่ง **for**

```
fruits = ["apple", "banana", "cherry"]  
for x in fruits:  
    if x == "banana":  
        break  
    print(x)
```



apple

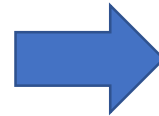
range() Function

- ฟังก์ชัน range() จะส่งคืนรายการค่าตัวเลข โดยเริ่มต้นจาก 0 (default) และเพิ่มค่าครั้งละ 1 (by default) จนถึงค่าตัวเลขที่ระบุไว้ ตัวอย่างเช่น

```
print(list(range(6)))
```

```
print(list(range(2,10)))
```

```
print(list(range(2,10,2)))
```



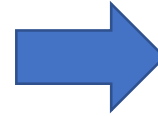
```
[0, 1, 2, 3, 4, 5]
```

```
[2, 3, 4, 5, 6, 7, 8, 9]
```

```
[2, 4, 6, 8]
```

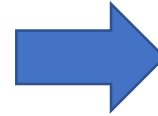
การใช้ range ในคำสั่ง for

```
for x in range(6):  
    print(x)
```



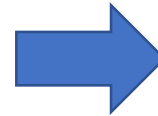
0
1
2
3
4
5

```
for x in range(2, 6):  
    print(x)
```



2
3
4
5

```
for x in range(2, 20, 3):  
    print(x)
```



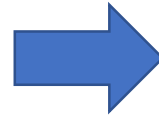
2
5
8
11
14
17

การใช้ range ร่วมกับ len ในคำสั่ง for

```
# Program to iterate through a list using indexing

genre = ['pop', 'rock', 'jazz']

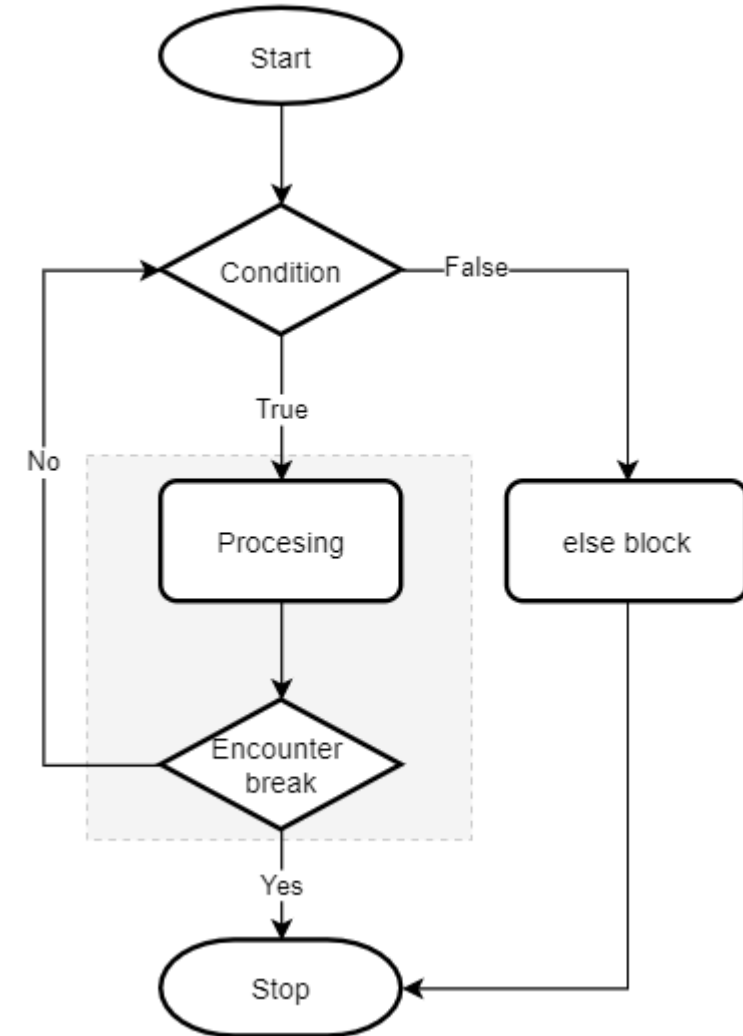
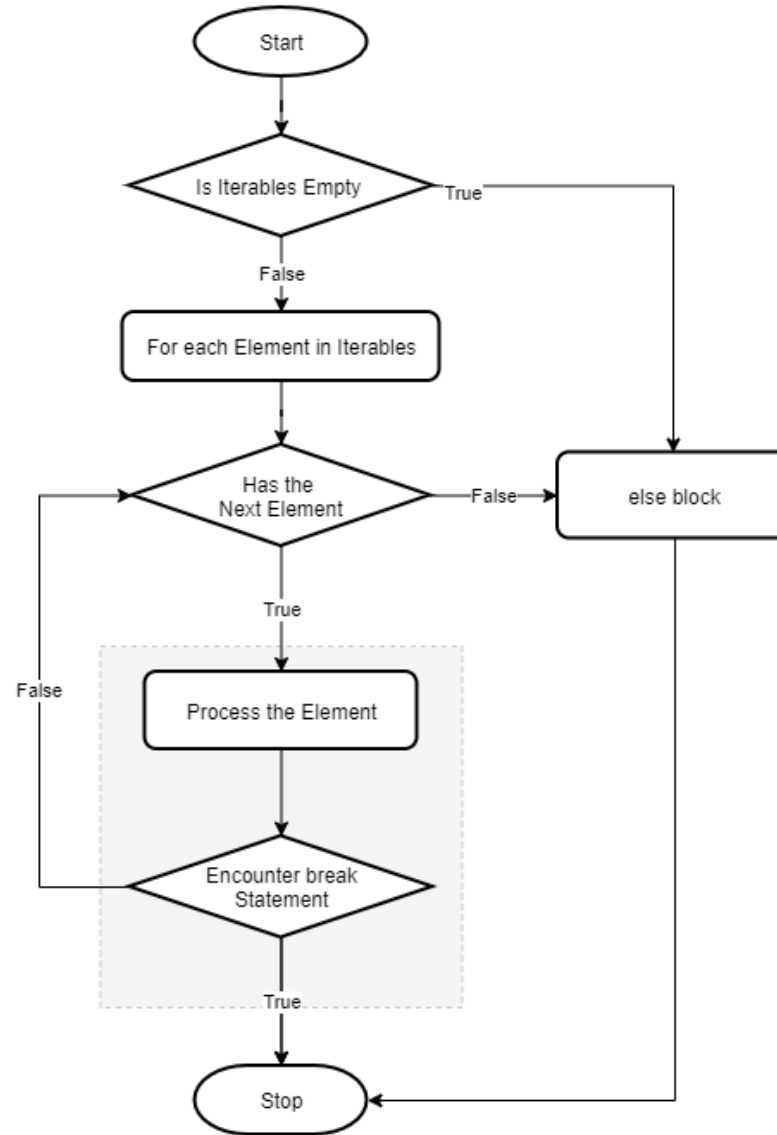
# iterate over the list using index
for i in range(len(genre)):
    print("I like", genre[i])
```



```
I like pop
I like rock
I like jazz
```


else...Loop

for...else หรือ while...else
Loops คือ การใช้ else ร่วมกับ
for หรือ while เพื่อรันคำสั่งเมื่อ
สิ้นสุดการทำงานของลูปวนซ้ำ
หรือ ไม่ได้วนซ้ำเลย



for...else...Loop

```
for item in iterables:
```

```
    # process item
```

```
else:
```

```
    # statement
```

```
for x in range(6):
```

```
    print(x)
```

```
else:
```

```
    print("Finally finished!")
```

1. ตารางสูตรผลคูณคูณจาก 1 ... 12 โดยรับค่าตัวเลขแม่คูณจากคีย์บอร์ด

```
try:  
    number = int(input("Enter a number: "))  
  
except ValueError:  
    print("กรุณากรอกค่าตัวเลขเท่านั้น")
```

```
Enter number : 5  
5 x 1 = 5  
5 x 2 = 10  
5 x 3 = 15  
5 x 4 = 20  
5 x 5 = 25  
5 x 6 = 30  
5 x 7 = 35  
5 x 8 = 40  
5 x 9 = 45  
5 x 10 = 50  
5 x 11 = 55  
5 x 12 = 60
```

2. จงเขียนโปรแกรมรับค่าคะแนนนิสิตจำนวน 10 คนและคำนวณผลรวมและค่าเฉลี่ยของคะแนนนิสิตทั้งหมด

```
try:

    for
        number = int(input("Enter a number: "))

except ValueError:
    print("กรุณารอกค่าตัวเลขเท่านั้น")
```

```
Enter score #1 : 85
Enter score #2 : 75
Enter score #3 : 65
Enter score #4 : 15
Enter score #5 : 65
Enter score #6 : 45
Enter score #7 : 90
Enter score #8 : 35
Enter score #9 : 75
Enter score #10 : 65
sum of score : 615
average score : 61.00
```

3. จงแสดงตัวเลขที่อยู่ในช่วงระหว่าง 0 ถึง 100 และหารด้วย 3 ลงตัว ว่ามีกี่ตัวเลข และเลขอะไรบ้าง

```
try:

    for

except ValueError:
    print("กรุณารอกค่าตัวเลขเท่านั้น")
```

```
0
3
6
9
12
15
18
21
24
27
30
33
36
39
42
45
48
51
54
57
60
63
66
69
72
75
78
81
84
87
90
93
96
99
number : 34
```

4. จงเขียนโปรแกรมทายตัวเลขสุ่มที่สร้างจากเครื่องคอมพิวเตอร์ในช่วง 0 ถึง 9 โดยกำหนดให้ทายจำนวน 3 ครั้ง

```
from numpy import random
number = random.randint(10)
try:
    for
        number = int(input("Enter a number: "))

except ValueError:
    print("กรุณากรอกค่าตัวเลขเท่านั้น")
```

```
Enter number #1 : 5
You are wrong !!!
Enter number #2 : 6
You are wrong !!!
Enter number #3 : 8
You are wrong !!!
The answer is 9.
```

ตัวอย่างโจทย์การใช้ else Loop

1. โปรแกรมรับจำนวนสินค้าจากคีย์บอร์ด แล้วจัดลงกล่องละ 12 ชิ้น ให้แสดงจำนวนกล่องทั้งหมดที่ต้องการใช้ โดยสามารถส่ง 3 รอบคือ เช้า กลางวัน และเย็น รอบละ 2 กล่องเท่านั้น จงแสดงจำนวนกล่องที่ต้องส่ง และหมายเลขกล่องจะต้องส่งในแต่ละรอบ

ตัวอย่างที่ 1

Enter number of goods : 36

Number of boxes : 3

วันที่ 1 เช้า ส่งกล่องหมายเลข 1 และ 2

วันที่ 1 กลางวัน ส่งกล่องหมายเลข 3

จบโปรแกรม

ตัวอย่างที่ 2

Enter number of goods : 100

Number of boxes : 9

วันที่ 1 เช้า ส่งกล่องหมายเลข 1 และ 2

วันที่ 1 กลางวัน ส่งกล่องหมายเลข 3 และ 4

วันที่ 1 เย็น ส่งกล่องหมายเลข 5 และ 6

วันที่ 2 เช้า ส่งกล่องหมายเลข 7 และ 8

วันที่ 2 กลางวัน ส่งกล่องหมายเลข 9

จบโปรแกรม

โครงสร้าง while

while condition :

statement1
statement2
statement3
...

Sum = 0

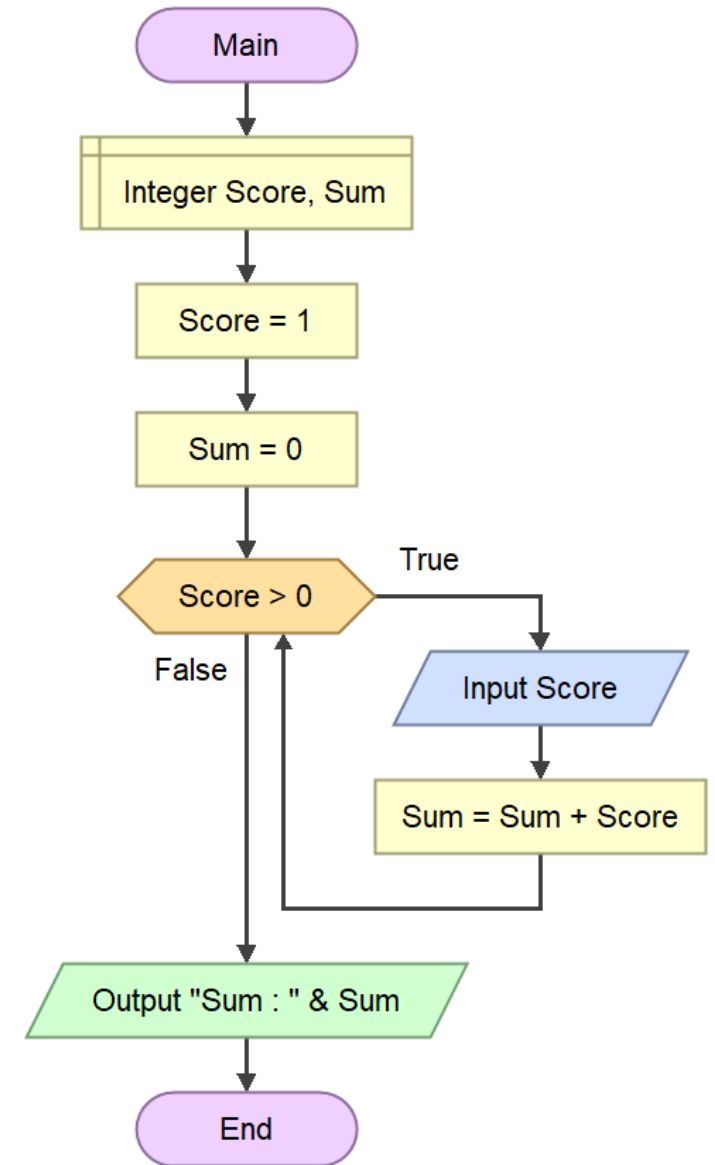
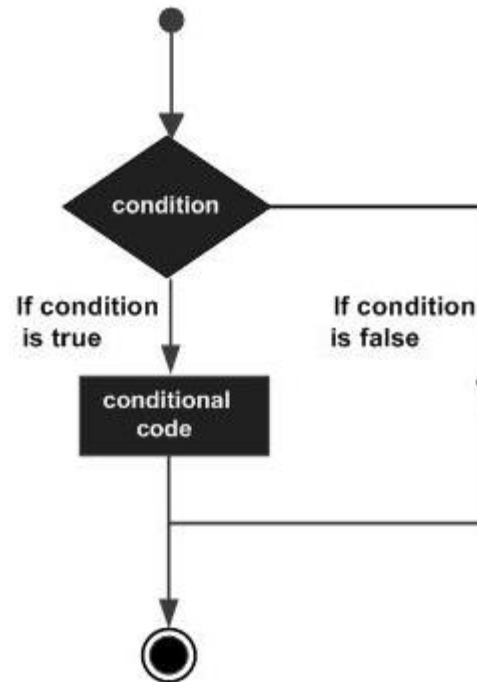
Score = 1

while Score > 0 :

Score = int(input("Enter score : "))

Sum = Sum + Score

print("sum = %d" %Sum)




```
ch = 'a'
while ch != 'z':
    print("character = %c" %ch)
    code = ord(ch) + 1 # next character code
    ch = chr(code)
```

```
odd = 0
num = int(input("Enter value: "))
while num > 0:
    odd = num % 2
    if (odd == 1) :
        print("num = %d" %num)
    num = num - 1
```

```
cont = 'y'
sum = 0
while cont.lower() == 'y':
    num = int(input("Enter value: "))
    sum = sum + num
    cont = input("continue (y/n) : ")
print("sum = %d" %sum)
```

```
i = 1
row = int(input("Enter row: "))
while row > 0:
    print("%s " %list(range(i,i + 4)))
    row -= 1
    i += 4
```

Infinite loop

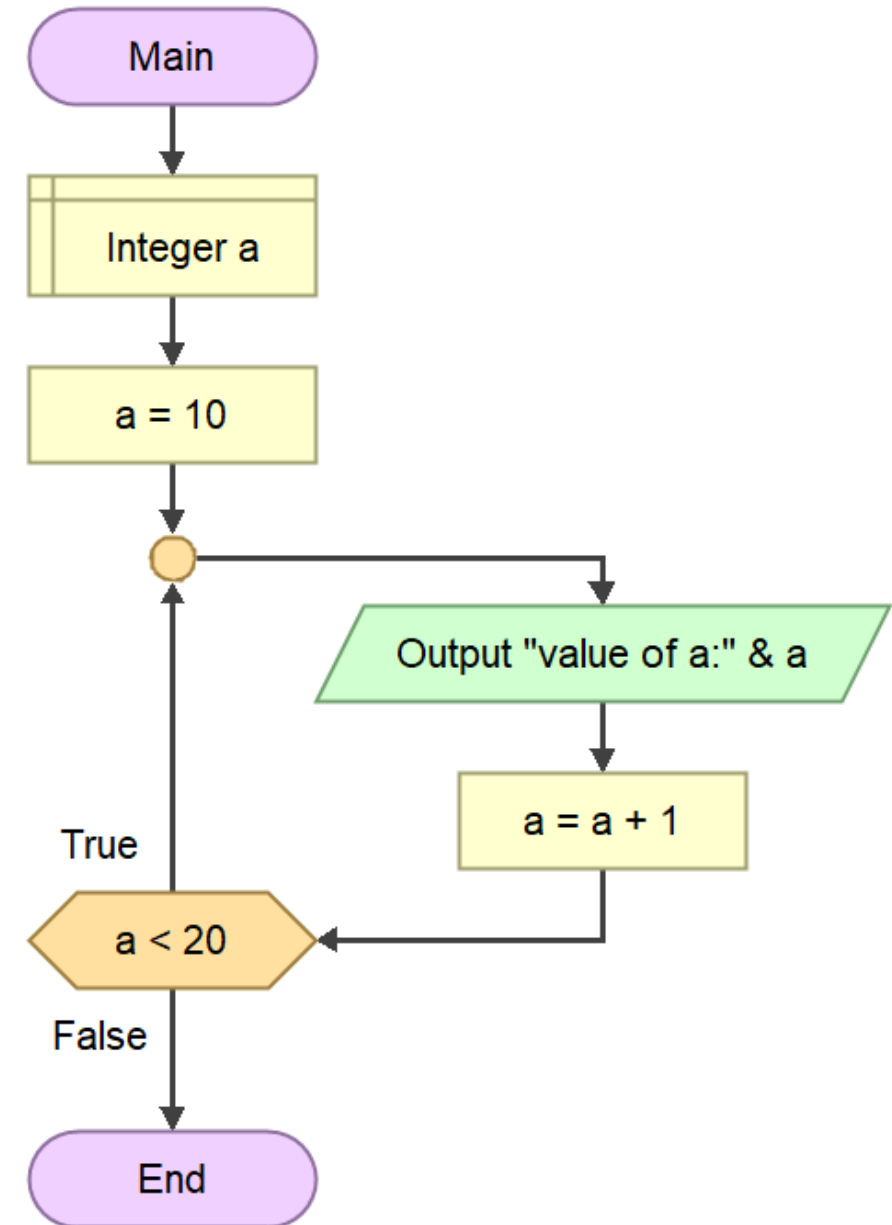
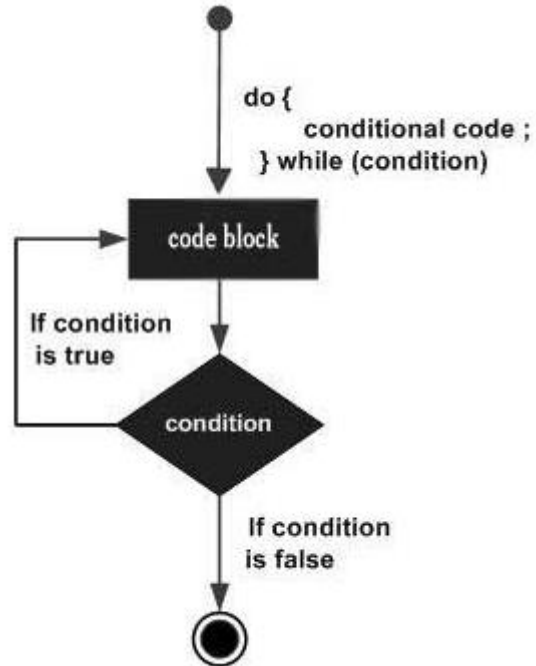
- ชุดคำสั่งรูป while ที่วนทำงานไปเรื่อย ๆ ไม่มีจุดสิ้นสุด
- ชุดคำสั่งที่ลบก่ียงหรือระมัดระวังใ้การใช้งาน เพราะจะก่อให้เกิดข้อผิดพลาด (bug) ในโปรแกรมหรือเครื่องคอมพิวเตอร์หยุดทำงาน (hang) ได้

```
while True:  
    statement1  
    statement2  
    statement3
```

โครงสร้าง DO...while

```
while True:  
    statement1  
    statement2  
    statement3  
    ...  
    if condition :  
        break
```

```
a = 10  
while True:  
    print("a = %d" %a)  
    if a >= 20 :  
        break  
    a = a + 1
```



```
ch = 'a'
while True:
    print("character = %c" %ch)
    code = ord(ch) + 1 # next character code
    ch = chr(code)
    if ch == 'z' :
        break
```

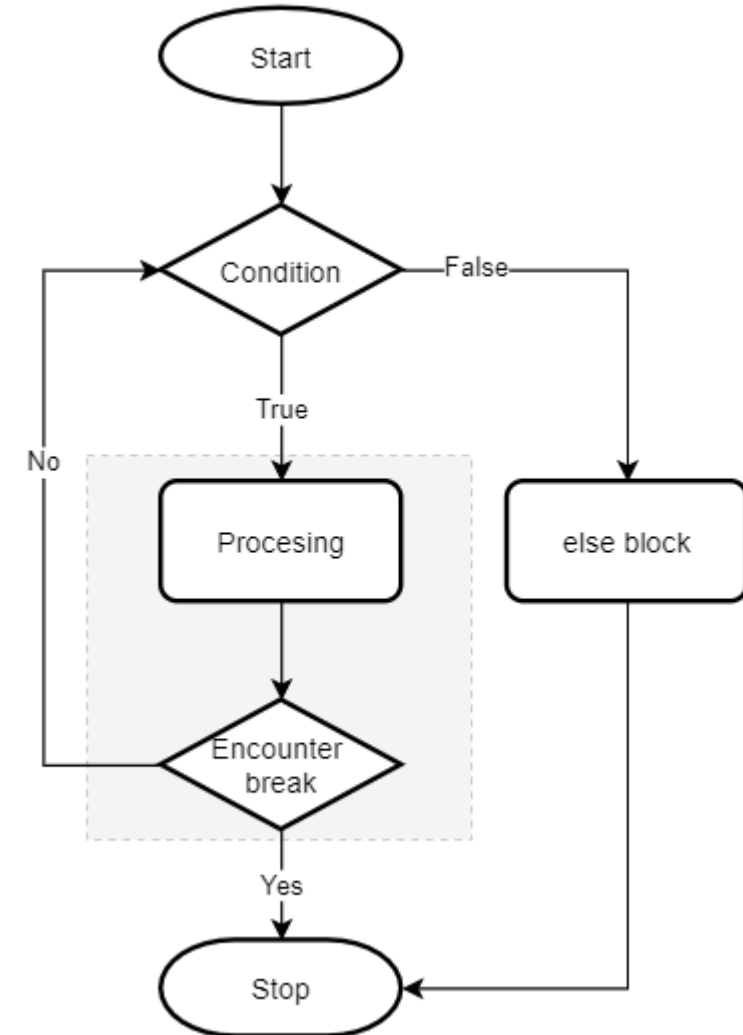
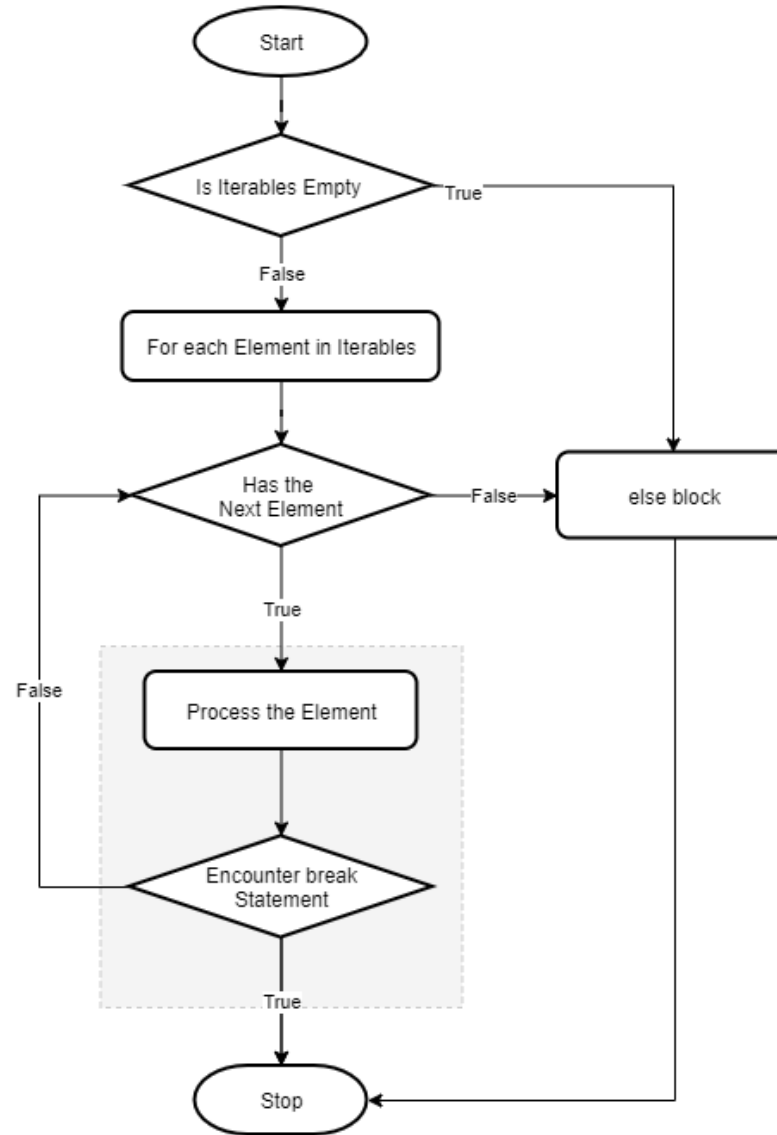
```
odd = 0
num = int(input("Enter value: "))
while True:
    odd = num % 2
    if (odd == 1) :
        print("num = %d" %num)
    num = num -1
    if num == 0 :
        break
```

```
cont = 'y'
sum = 0
while True:
    num = int(input("Enter value: "))
    sum = sum + num
    cont = input("continue (y/n) : ")
    if cont.lower() == 'n' :
        break
print("sum = %d" %sum)
```

```
i = 1
row = int(input("Enter row: "))
while True:
    print("%s " %list(range(i,i + 4)))
    row -= 1
    if row == 0 :
        break
    i += 4
```

else...Loop

for...else หรือ while...else
Loops คือ การใช้ else ร่วมกับ
for หรือ while เพื่อรันคำสั่งเมื่อ
สิ้นสุดการทำงานของลูปวนซ้ำ
หรือ ไม่ได้วนซ้ำเลย



while...else...Loop

while condition:

code block to run

else:

else clause code block

i = 6

while i > 0:

print (i)

i -= 1

else:

print("Finally finished!")

Nested Loops

Nested Loops คือ การใช้ for หรือ while ซ้อนทำงานกัน

```
for i in range (2) :
```

```
    print ("row : " , i)
```

```
        for j in range (5) :
```

```
            print (i, j)
```

```
else:
```

```
    print("-----END-----")
```

```
i = 1
```

```
while i <= 2:
```

```
    print ("row : " , i)
```

```
        for j in range (5) :
```

```
            print (i, j)
```

```
else :
```

```
    print("-----END-----")
```

ตัวอย่างโจทย์การใช้ Nested Loop

1. โปรแกรมแสดงตารางสูตรผลคูณตั้งแต่แม่ 2 ถึง 5 โดยแสดงผลคูณจาก 1 ... 12

ตัวอย่าง

$$1 \times 1 = 1$$

$$1 \times 2 = 2$$

.

$$1 \times 12 = 12$$

...

$$5 \times 1 = 5$$

$$5 \times 2 = 10$$

.

$$5 \times 12 = 60$$

2. โปรแกรมรับออเดอร์ลูกค้าก็ได้ จนกระทั่งเลือก No ในแต่ล่ออเดอร์รับรายการสั่งซื้อเท่าไรก็ได้จนกว่าเลือก 0

ตัวอย่าง

Menu

1. Coke 20 baht
2. Milk 15 baht
3. Tea 20 baht
4. Water 10 baht

Order #1

Item : 1

Item : 4

Item : 0

Amount : 30 baht

Continue (Y/N) : Y

Order #2

Item : 2

Item : 3

Item : 0

Amount : 35 baht

Continue (Y/N) : N

แบบฝึกหัด

1. ตารางสูตรผลคูณจาก 1 ... 12 โดยรับค่าตัวเลขแม่คูณจากคีย์บอร์ด

ตัวอย่าง

Enter Number : 2

$$2 \times 1 = 2$$

$$2 \times 2 = 4$$

.

$$2 \times 12 = 24$$

2. จงเขียนโปรแกรมรับค่าคะแนนนิสิตที่คนก็ได้จนกระทั่งใส่คะแนนติดลบ และคำนวณผลรวมและค่าเฉลี่ยของคะแนนนิสิตทั้งหมด

ตัวอย่าง

Enter Score #1 : 70

Enter Score #2 : 85

Enter Score #3 : 63

.

Enter Score #16 : -1

Sum of Score : 1125

Average Score : 75

แบบฝึกหัด

3. จงเขียนโปรแกรมทายตัวเลขสุ่มที่สร้างจากเครื่องคอมพิวเตอร์ในช่วง 0 ถึง 9 โดยกำหนดให้ทายจนกระทั่งทายถูก แล้วแสดงจำนวนครั้งที่ทายไป

ตัวอย่างที่ 1

Enter Number : 7

You are wrong !!!

Enter Number : 8

Your are wrong !!!

Enter Score : 3

Your are right !!!

The number of trial is 5

ตัวอย่างโจทย์การใช้ else Loop

2. โปรแกรมนับรถยนต์และจักรยานยนต์ของที่รับฝากรถ
แห่งหนึ่ง ที่รองรับรถยนต์ได้ 10 คัน และจักรยานยนต์ 50
คัน

- รับจำนวนรถยนต์และจักรยานยนต์ที่ต้องการจอดจาก
คีย์บอร์ดจนกระทั่งใส่จำนวนรถยนต์และจักรยานยนต์
เท่ากับ 0
- คำนวณที่จอดรถยนต์และจักรยานยนต์ที่เหลือว่างอยู่
- ถ้าที่จอดรถเต็มให้แสดงว่า Parking lot is full. และ
หยุดโปรแกรม

ตัวอย่างที่ 1

cars : 1

motorcycles : 4

Remaining lots cars : 9 and motorcycles: 46

....

cars : 0

motorcycles : 0

Stop Program !!!

ตัวอย่างที่ 2

cars : 10

Motorcycles : 50

Parking lot is full !!!

Stop Program !!!