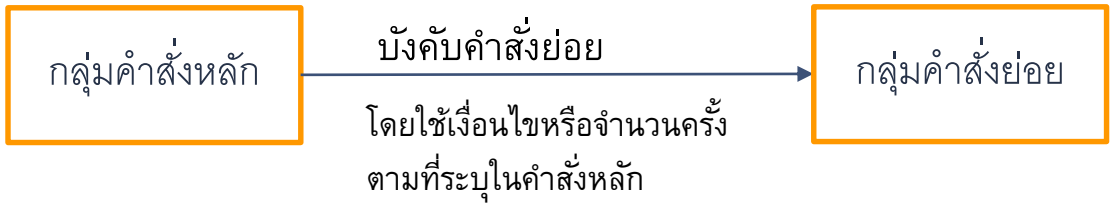
Iteration
(while loop) การวนซ้ำ

การวนซ้ำ (Iteration / Loop)

พื้นฐานการเขียนโปรแกรม



เรียกคำสั่งในลักษณะนี้ว่า **คำสั่งการวนซ้ำ**

- คำสั่ง for
- คำสั่ง while

การวนซ้ำ (Iteration / Loop)

การทำงานของโปรแกรมซึ่งมีการทำงานบางส่วนที่ซ้ำกันหลายครั้ง เช่น

- โปรแกรมคำนวณเกรดเฉลี่ยของนักเรียนแต่ละคน
- โปรแกรมคำนวณราคารวมของสินค้าที่ต้องชำระ

การวนซ้ำมีส่วนประกอบหลัก 2 ส่วนคือ

- เงื่อนไขการทำซ้ำ (loop condition)
- งานส่วนที่ต้องทำซ้ำ (loop body)

Outline

Loop

- while
- break
- continue

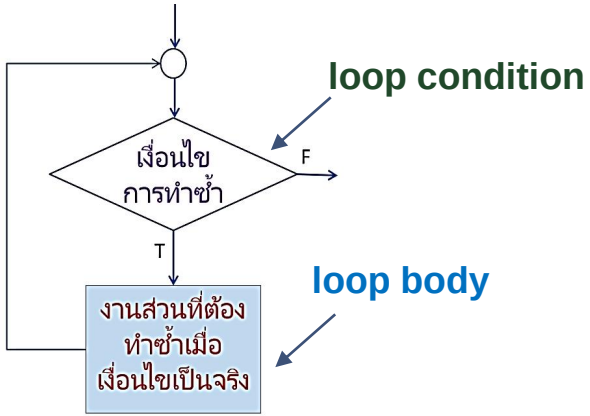
1

while

คำสั่ง while

- เป็นคำสั่งแบบวนซ้ำ
- จะทำคำสั่งที่ต้องทำซ้ำภายในลูป (loop body)

  ↳ ถ้าตรวจสอบเงื่อนไขการทำซ้ำหลัง while แล้วเป็น ***จริง***



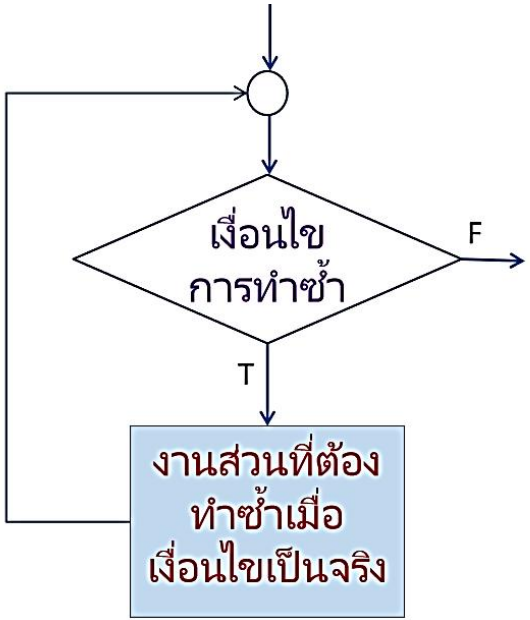
คำสั่ง while

ตัวอย่าง

- Loop n times
- Loop until a specific event occurs
- Loop until a specific event occurs n times

คำสั่ง while

- เป็นคำสั่งแบบวนซ้ำ จะทำคำสั่งที่ย่อยที่ ต้องการวนซ้ำภายในลูป (loop body)
- ถ้าตรวจสอบเงื่อนไขการทำซ้ำหลัง while แล้วเป็น**จริง**



รูปแบบคำสั่งคือ

while เงื่อนไขการทำซ้ำ:

ย่อหน้า indent

งานส่วนที่ต้องทำซ้ำเมื่อเงื่อนไขเป็นจริง

(loop body)

คำสั่งถัดไปต่อจากลูป

คำสั่ง while

รูปแบบคำสั่งคือ

```
while เงื่อนไขการทำซ้ำ:
    งานส่วนที่ต้องทำซ้ำเมื่อเงื่อนไขเป็นจริง
    (loop body)
คำสั่งถัดไปต่อจากลูป
```

ย่อหน้า indent

- คล้าย if ใช้ indent เพื่อกั้นขอบเขตของ block คำสั่งในส่วนของ loop body
- ถ้าเงื่อนไขเป็นเท็จจะจบลูปแล้วทำคำสั่งถัดไปต่อจากลูปคำสั่งนี้ โดยคำสั่งถัดไปจะเขียนเยื้องย่อหน้าให้ตรงกับ while
- ใน loop body จะต้องมีคำสั่งเปลี่ยนค่าตัวแปรที่ใช้ในเงื่อนไขการทำซ้ำ เพื่อให้พบกรณีที่เงื่อนไขเป็นเท็จและหยุดลูปได้
- หากเงื่อนไขเป็นจริงตลอดลูปจะวนซ้ำไม่รู้จบ (infinite loop)

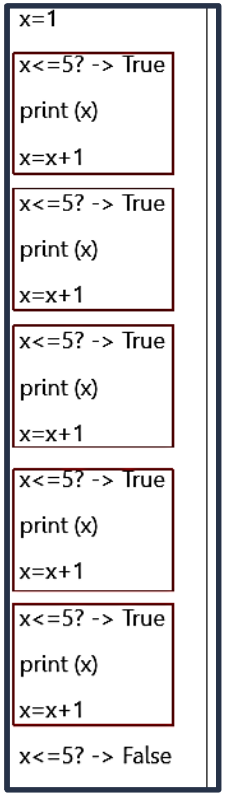
Loop n times

ตัวอย่าง : แสดงค่าของเลข 1-5 เป็นผลลัพธ์

```
print (1)
print (2)
print (3)
print (4)
print (5)
```

```
x=1
print (x)
x=2
print (x)
x=3
print (x)
x=4
print (x)
x=5
print (x)
```

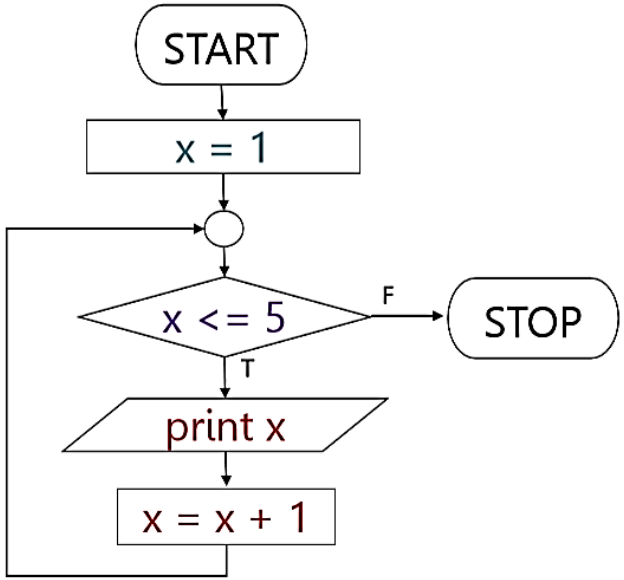
```
x=1
print (x)
x=x+1
print (x)
x=x+1
print (x)
x=x+1
print (x)
x=x+1
print (x)
```




```
x=1
while x<=5:
    print (x)
    x=x+1
```

Loop n times

ตัวอย่าง : แสดงค่าของเลข 1-5 เป็นผลลัพธ์



Idea

Variable assignment (Initial state before while) : กำหนดตัวแปรเริ่มต้น สมมติให้ x=1

Process: ตรวจสอบเงื่อนไขว่าค่า x น้อยกว่าหรือเท่ากับ 5
โดยใช้ เงื่อนไข while ตรวจสอบ
ถ้า x<=5 เป็นจริง

- แสดงค่า x ออกมาทางหน้าจอโดยใช้คำสั่ง print x
- จากนั้นทำการเพิ่มค่าตัวแปร x ทีละ 1 โดยใช้คำสั่ง x=x+1 หรือ x+=1
- ส่งค่า x วนขึ้นไปตรวจสอบเงื่อนไขที่ while loop

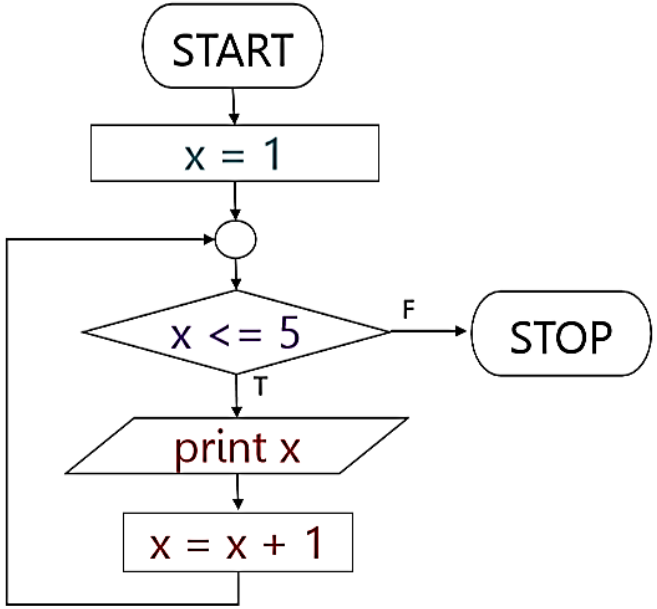
- โปรแกรมจะวนทำงานไปเรื่อยๆ จนเจอเงื่อนไขที่เป็นเท็จแล้วโปรแกรมจะจบการวน loop

ถ้า x<=5 เป็นเท็จ จบการทำงาน

Output: แสดงผลค่า x

Loop n times

ตัวอย่าง : แสดงค่าของเลข 1 - 5 เป็นผลลัพธ์



PYTHON CODE

```
x = 1
while x <= 5:
    print (x)
    x = x + 1
```

ผลที่ได้คือ

1
2
3
4
5

Loop n times

ตัวอย่าง : แสดงค่าของเลข 1–5 เป็นผลลัพธ์

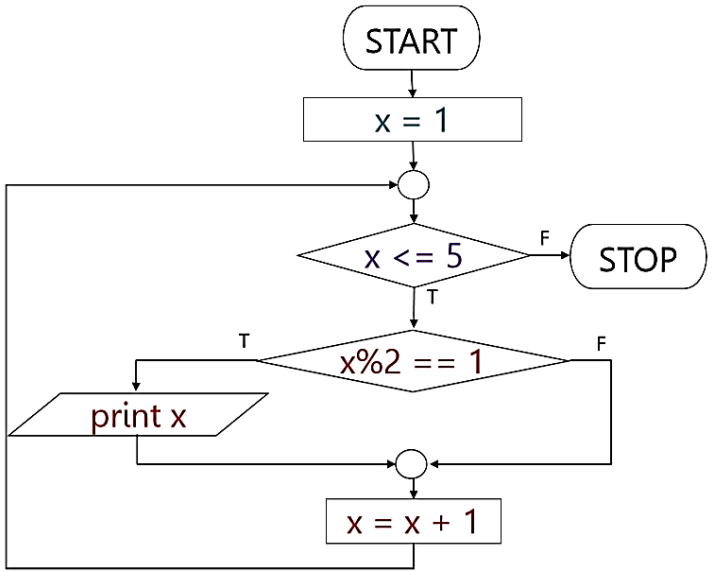
```
x = 1
while x <= 5:
    print (x)
    x = x + 1
```

ตารางแสดงผลการทดสอบเงื่อนไข และค่าภายในตัวแปร

การทดสอบการวนซ้ำครั้งที่	ผลการทดสอบเงื่อนไขการวนซ้ำ x<=5	ผลลัพธ์ทางจอภาพ	ค่าของตัวแปรที่ x เปลี่ยนใน loop body (จากที่กำหนด x=x+1)
1	True	1	2
2	True	2	3
3	True	3	4
4	True	4	5
5	True	5	6
6	False		

Loop n times

ตัวอย่าง : แสดงเฉพาะค่าของเลขคี่ที่อยู่ในช่วง 1- 5 เป็นผลลัพธ์



Idea

Variable assignment (Initial state before while) : กำหนดตัวแปรเริ่มต้น สมมติให้ x=1

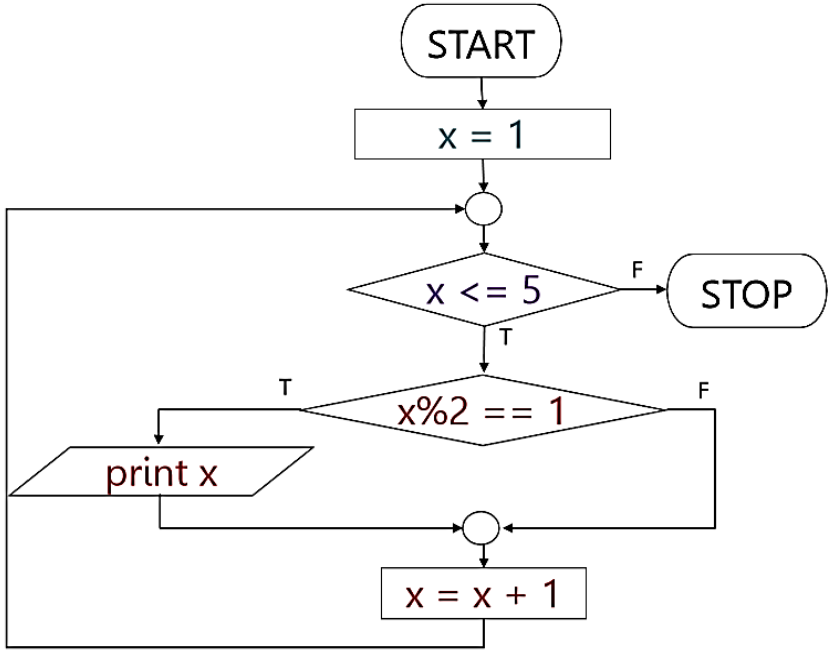
Process: ตรวจสอบเงื่อนไขว่าค่า x น้อยกว่าหรือเท่ากับ 5
โดยใช้ เงื่อนไข while ตรวจสอบ
ถ้า x<=5 เป็นจริง
- ตรวจสอบเงื่อนไขว่าเป็นเลขคี่หรือไม่โดย ใช้เงื่อนไข if ตรวจสอบ
  -- เงื่อนไขเพื่อตรวจสอบเลขคี่ x%2==1
  -- ถ้า x%2==1 เป็นจริง แสดงผลลัพธ์ค่าตัวเลขนั้นออกจากทางหน้าจอ
  -- ถ้า x%2==1 เป็นเท็จ
  -- หลังตรวจสอบเงื่อนไข if จากนั้นทำการเพิ่มค่าตัวแปร x ทีละ 1 โดยใช้คำสั่ง x=x+1
  -- นำค่า x วนขึ้นไปตรวจสอบเงื่อนไขที่ while loop

ถ้า x<=5 เป็นเท็จ จบการทำงาน

Output: แสดงผลค่า x ที่เป็นเลขคี่

Loop n times

ตัวอย่าง : แสดงเฉพาะค่าของเลขคี่ที่อยู่ในช่วง 1- 5 เป็นผลลัพธ์



PYTHON CODE

```
x = 1
while x <= 5:
    if x % 2 == 1:
        print (x)
    x = x + 1
```

ผลที่ได้คือ

1

3

5

Loop n times

ตัวอย่าง : แสดงค่าของเลข 1–5 เป็นผลลัพธ์

```
x = 1
while x <= 5:
    if x % 2 == 1:
        print (x)
    x = x + 1
```

ตารางแสดงผลการทดสอบเงื่อนไข และค่าภายในตัวแปร

การทดสอบการวนซ้ำครั้งที่	ผลการทดสอบเงื่อนไขการวนซ้ำ x<=5	ผลการทดสอบเงื่อนไขทางเลือก x%2==1	ผลลัพธ์ทางจอภาพ	ค่าของตัวแปรที่ x เปลี่ยนใน loop body (จากที่กำหนด x=x+1)
1	True	True	1	2
2	True	False		3
3	True	True	3	4
4	True	False		5
5	True	True	5	6
6	False			

Loop n times

ตัวอย่าง : แสดงค่าของเลข 1–5 เป็นผลลัพธ์

```
x = 1
while x <= 5:
    if x % 2 == 1:
        print (x)
    x = x + 1
```

หรือเขียนได้อีกแบบเป็น

```
x = 1
while x <= 5:
    print (x)
    x = x + 2
```

Quiz

Quiz

คำนวณผลรวมของจำนวนเต็มตั้งแต่ 1 ถึง 100 แสดงเป็นผลลัพธ์

คำถามก่อนจะเริ่มเขียนโปรแกรม

- ลูปนี้จะทำงานกี่ครั้ง
- ใช้กี่ตัวแปร
- ตัวแปรใดเป็นตัวแปรเช็คเงื่อนไขของลูป
- เปลี่ยนค่าตัวแปรที่ใช้เช็คเงื่อนไขของลูปอย่างไร
- ลองเขียนคำสั่งที่ต้องใช้

Solution Quiz

คำนวณผลรวมของจำนวนเต็มตั้งแต่ 1 ถึง 100 แสดงเป็นผลลัพธ์

Idea

Variable assignment (Initial state before while) :
กำหนดตัวแปรเริ่มต้น สำหรับตัวแปร x และ
ตัวแปรสำหรับเก็บผลรวมค่า x โดยสมมติให้ x=1 และ sumv=0

Process: ตรวจสอบเงื่อนไขว่าค่า x น้อยกว่าหรือเท่ากับ 100
โดยใช้ เงื่อนไข while ตรวจสอบ
ถ้า x<=100 เป็นจริง
- คำนวณผลรวมของตัวเลขแล้วเก็บไว้ที่ตัวแปร sumv
- ทำการเพิ่มค่าตัวแปร x ทีละ 1 โดยใช้คำสั่ง x=x+1
- หลังจากนั้นนำค่า x วนไปตรวจสอบที่เงื่อนไข while

ถ้า x<=100 เป็นเท็จ จบการทำงาน

Output: แสดงผลรวมของจำนวนเต็มตั้งแต่ 1-100

PYTHON CODE

```
x=1
sumv=0

while x<=100:
    sumv=sumv+x
    x=x+1
print('summation of 1-100:', sumv)
```

Quiz

เขียนโปรแกรมเพื่อแสดงค่าสูตรคูณแม่ 7 โดยแสดงผลลัพธ์ที่ได้ตามจอภาพ

ผลลัพธ์ที่แสดงบนจอภาพ คือ

7*1 = 7

7*2 = 14

7*3 = 21

.....

7*11 = 77

7*12 = 84

end

คำถามก่อนจะเริ่มเขียนโปรแกรม

- ลูปนี้จะทำงานกี่ครั้ง
- ใช้กี่ตัวแปร
- ตัวแปรใดเป็นตัวแปรเช็คเงื่อนไขของลูป
- เปลี่ยนค่าตัวแปรที่ใช้เช็คเงื่อนไขของลูปอย่างไร
- ลองเขียนคำสั่งที่ต้องใช้

Solution Quiz (try it yourself)

เขียนโปรแกรมเพื่อแสดงค่าสูตรคูณแม่ 7 โดยแสดงผลลัพธ์ที่ได้ตามจอภาพ

Idea

Variable assignment (Initial state before while) :

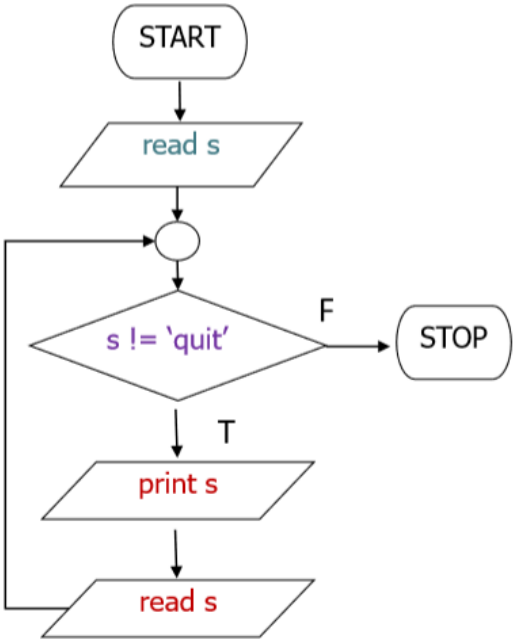
Process:

Output:

PYTHON CODE

Loop until a specific event occurs

ตัวอย่าง : วนรับสายอักขระและแสดงผลเป็นสายอักขระนั้น โดยให้หยุด การทำงานเมื่อสายอักขระที่รับมาเป็นคำว่า quit



Idea

Input (Initial state before while) : ให้โปรแกรมรับสตริง s

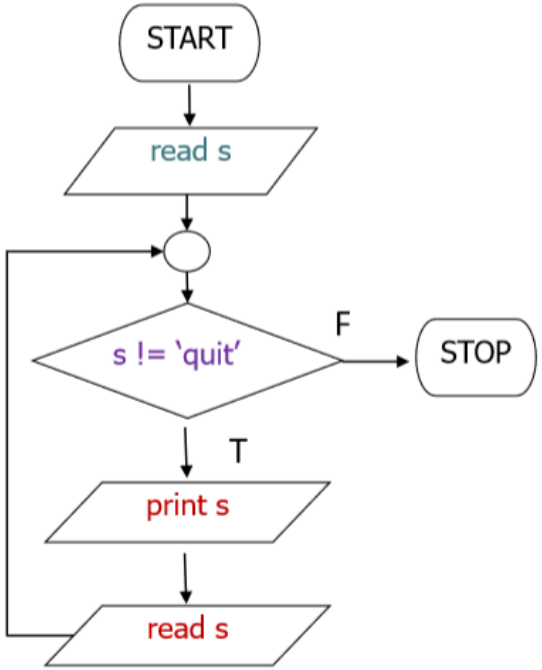
Process: ตรวจสอบเงื่อนไขว่าสตริง s เป็นคำว่า 'quit' หรือไม่
โดยใช้ เงื่อนไข while ตรวจสอบ
ถ้า s!='quit' เป็นจริง
- แสดงผลลัพธ์ออกจากจอภาพ
- จากนั้นให้มีการรับสตริง s อีกครั้ง (เป็นการรับสตริงใต้ลูป)
- หลังจากนั้นนำสตริง s วนขึ้นไปตรวจสอบที่เงื่อนไข while

ถ้า s!='quit' เป็นเท็จ จบการทำงาน (ความหมายคือ รับ s เป็นคำว่า 'quit' พอดี)

Output: แสดงสายอักขระ

Loop until a specific event occurs

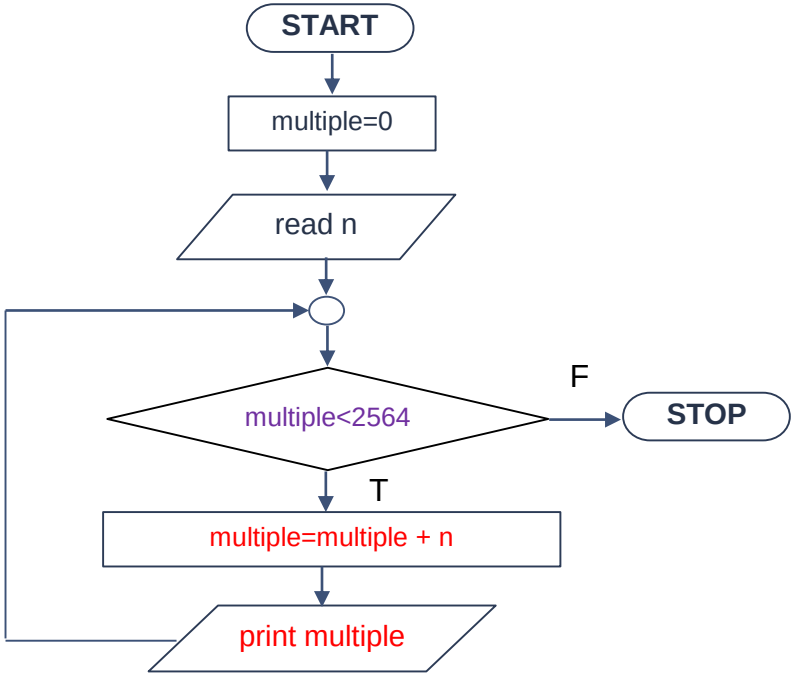
ตัวอย่าง : วนรับสายอักขระและแสดงผลเป็นสายอักขระนั้น โดยให้หยุด การทำงานเมื่อสายอักขระที่รับมาเป็นคำว่า quit




```
s = input('Enter a string : ')
while s != 'quit':
    print ('You enter : ', s)
    s = input('Enter a string : ')
```

Loop until a specific event occurs

ตัวอย่าง : รับค่าเลขจำนวนเต็ม 1 จำนวนแล้วหาผลคูณของเลขจำนวนนั้นซึ่งมีค่าน้อยกว่าค่า 2564



Idea

Input (Initial state before while) :
ให้โปรแกรมรับเลขจำนวนเต็ม 1 ค่า เก็บไว้ที่ตัวแปร n
กำหนดค่าเริ่มต้นของตัวแปรที่จะนำมาเก็บผลคูณ multiple=0

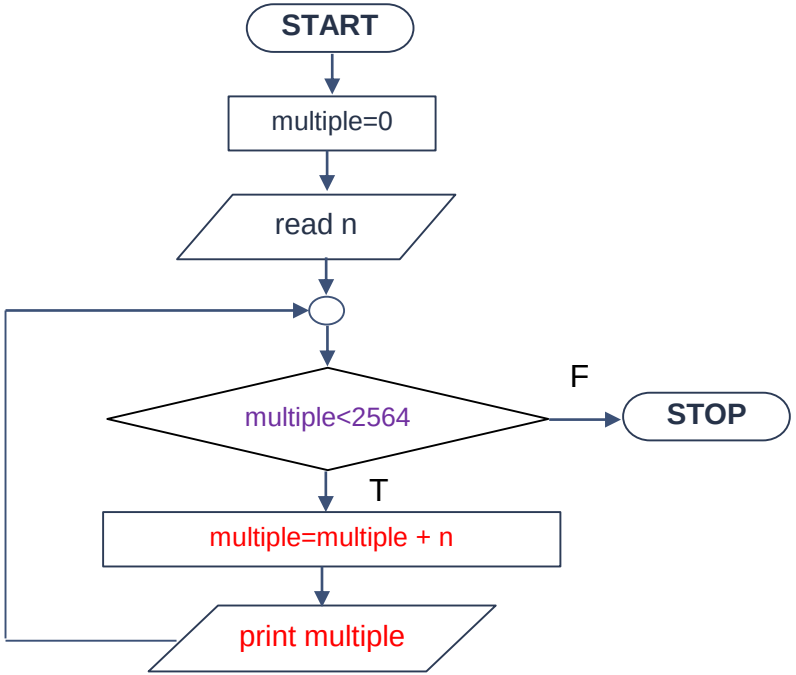
Process: ตรวจสอบเงื่อนไขว่าค่า multiple น้อยกว่า 2564 หรือไม่
โดยใช้ เงื่อนไข while ตรวจสอบ
ถ้า multiple < 2564 เป็นจริง
- แสดงผลลัพธ์ออกจากจอภาพ

ถ้า multiple < 2564 เป็นเท็จ จบการทำงาน

Output: แสดงผลคูณของเลขจำนวนนั้นซึ่งมีค่าน้อยกว่า 2564

Loop until a specific event occurs

ตัวอย่าง : รับค่าเลขจำนวนเต็ม 1 จำนวนแล้วหาผลคูณของเลขจำนวนนั้นซึ่งมีค่าน้อยกว่าค่า 2564

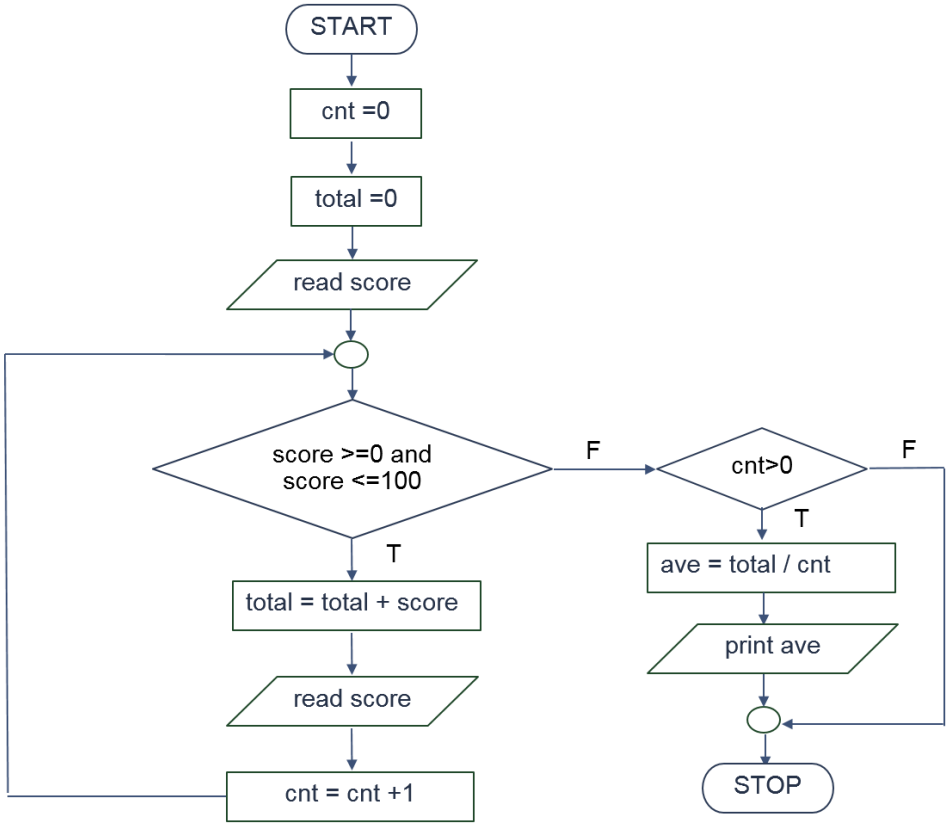


PYTHON CODE

```
multiple=0
n=int(input('Enter:'))
while multiple<2564:
  multiple = multiple + n
  print(multiple)
```

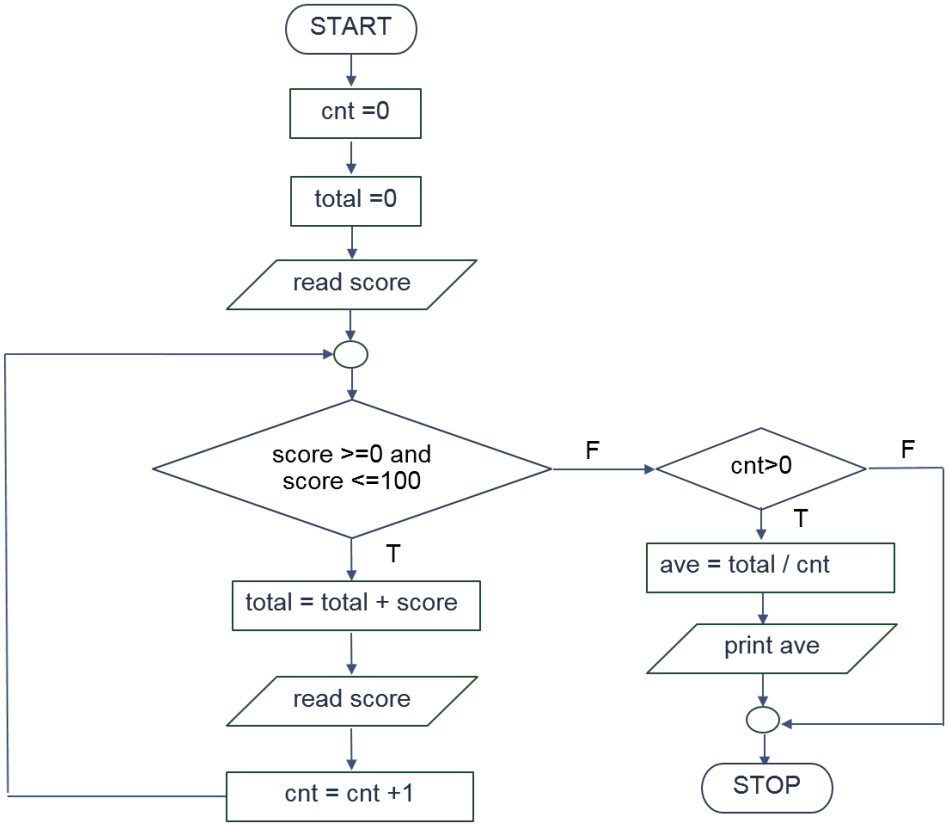
Loop until a specific event occurs

ตัวอย่าง : วนรับค่าคะแนนสอบจนกว่าคะแนนจะไม่อยู่ในช่วง 0-100 แล้วหาค่าเฉลี่ยของคะแนนแสดงเป็นผลลัพธ์



Loop until a specific event occurs

ตัวอย่าง : วนรับค่าคะแนนสอบจนกว่าคะแนนจะไม่อยู่ในช่วง 0-100 แล้วหาค่าเฉลี่ยของคะแนนแสดงเป็นผลลัพธ์



PYTHON CODE

```
cnt = 0
total = 0
score = int (input('enter score : '))
while score>=0 and score <=100:
    total = total +score
    score = int (input('enter score : '))
    cnt = cnt+1
if cnt>0:
    ave = total /cnt
    print ('average score : ', ave)
```

Quiz

Quiz

จงเขียนผังงานและโปรแกรมเพื่อวนรับค่าเลขจำนวน เต็ม 2 จำนวน แล้วแสดงผลรวมของเลขทั้งสองเป็น ผลลัพธ์ จนกว่าผลรวมของเลขทั้งสองน้อยกว่า 5

ตัวอย่างการรันโปรแกรม

Enter a : 3<enter>
Enter b : 5<enter>
8

Enter a : 5<enter>
Enter b : 7<enter>
12

Enter a : 1<enter>
Enter b : 2<enter>

Solution Quiz

จงเขียนผังงานและโปรแกรมเพื่อวนรับค่าเลขจำนวน เต็ม 2 จำนวน แล้วแสดงผลรวมของเลขทั้งสองเป็น ผลลัพธ์ จนกว่าผลรวมของเลขทั้งสองน้อยกว่า 5

ตัวอย่างการรันโปรแกรม

Enter a : 3<enter>
Enter b : 5<enter>
8

Enter a : 5<enter>
Enter b : 7<enter>
12

Enter a : 1<enter>
Enter b : 2<enter>

PYTHON CODE

```
a=int(input('Enter a:'))
b=int(input('Enter b:'))
while a+b>5:
    print(a+b)
    a=int(input('Enter a:'))
    b=int(input('Enter b:'))
```

Loop until a specific event occurs n times

***ตัวอย่าง :** วนรับค่าอายุ จนกว่าจะได้อาสาสมัครที่มีอายุตั้งแต่ 20 ปีขึ้นไปครบ 3 คน

ถ้าใครยังเขียนเขียนโปรแกรมไม่ได้ ลองนึกคำสั่งได้ประมาณนี้??

```
age = int(input("Enter age : "))
if age >= 20:
n = 0
while n < 3:
n = n + 1
print("Now we have 3 volunteers with age>=20")
```

ลองเอาชุดคำสั่งข้างบนมาจัดเรียงและย่อหน้าให้เป็นโปรแกรมที่ถูกต้อง

Loop until a specific event occurs n times

ตัวอย่าง : วนรับค่าอายุ จนกว่าจะได้อาสาสมัครที่มีอายุตั้งแต่ 20 ปีขึ้นไปครบ 3คน

```
n = 0
while n < 3:
  age = int(input("Enter age : "))
  if age >= 20:
    n = n + 1
print("Now we have 3 volunteers with age>=20")
```

ตัวอย่างการรันโปรแกรม

```
Enter age : 19<enter>
Enter age : 33<enter>
Enter age : 11<enter>
Enter age : 22<enter>
Enter age : 12<enter>
Enter age : 44<enter>
Now we have 3 volunteers with age>=20
```

2

break

คำสั่ง break

- เป็นคำสั่งที่เขียนอยู่ภายใน loop body ซึ่งจะทำให้หยุดการวนซ้ำทันทีและไม่ประมวลผลคำสั่งอื่นที่เหลืออยู่ใน loop body อีก
- จะมีการตรวจสอบเงื่อนไขด้วยคำสั่ง if เพื่อสั่งให้ break ทำงานภายใต้เงื่อนไขที่กำหนด
- หากเป็น loop ซ้อน loop คำสั่ง break จะทำให้หลุดจาก loop ชั้น ที่ครอบคำสั่ง break อยู่เท่านั้น

break

ตัวอย่าง : แสดงค่าตัวเลขตั้งแต่ 5 ลงมาถึง 1 โดยให้มีการใช้คำสั่ง break เพื่อหยุดการวนซ้ำเมื่อตัวเลขมีค่าเป็น 3 นั่นคือจะพิมพ์ได้แค่เลข 5, 4 เท่านั้น

```
x = 5
while x>=1:
    if x == 3:
        break
    print(x)
    x = x – 1
```

ผลที่ได้คือ

5

4

โปรแกรมนี้ให้ผลเหมือนกันหรือไม่

```
x = 5
while True:
    if x == 3:
        break
    print(x)
    x = x - 1
```

break

ตัวอย่าง : วนรับสายอักขระและแสดงผลเป็นสายอักขระนั้น โดยให้หยุดการทำงานเมื่อสายอักขระที่รับมาเป็นคำว่า quit

```
s = input('Enter a string : ')
while s != 'quit':
    print ('You enter : ', s)
    s = input('Enter a string : ')
```

ผลที่ได้คือ

```
Enter a string : Hi<enter>
You enter :  Hi
Enter a string : hello<enter>
You enter :  hello
Enter a string : quit<enter>
```

โปรแกรมนี้ให้ผลเหมือนกันหรือไม่

```
while True:
    s = input('Enter a string : ')
    if s == 'quit':
        break
    print ('You enter : ', s)
```

3

continue

คำสั่ง continue

- จะสั่งให้การทำงานย้อนกลับไปที่ส่วนต้นของการวนซ้ำทันทีโดยไม่สนใจคำสั่งอื่นที่เหลืออยู่ภายใน loop body
- นั่นคือจะข้ามคำสั่งที่เหลืออยู่และเริ่มต้นรอบใหม่ของการวนซ้ำ
- ในทำนองเดียวกันกับคำสั่ง break หากมี loop ซ้อน loop คำสั่ง continue จะทำให้ย้อนกลับไปที่ตอนต้นของ loop ชั้นที่ครอบคำสั่ง continue อยู่เท่านั้น

continue

ตัวอย่าง : แสดงค่าตัวเลขตั้งแต่ 5 ลงมาถึง 1โดยให้มีการใช้คำ สั่ง continue เพื่อข้ามการพิมพ์เลข 3

```
x = 6
while x > 1:
    x = x - 1
    if x == 3:
        continue
    print (x)
```

ผลที่ได้คือ

```
5
4
2
1
```

เขียนแบบนี้ได้หรือไม่

```
x = 5
while x > 0:
        if x == 3:
            continue
        print (x)
        x = x - 1
```

Exercise

แบบฝึกหัด

จงเติมช่องว่างในโปรแกรมวนรับสายอักขระเป็นชื่อของสายพันธ์ Covid-19 โดยให้หยุดการทำงานเมื่อสายอักขระที่รับมาเป็นคำว่า Omicron
ตรวจสอบเงื่อนไขด้วย while loop

PYTHON CODE:

```
Cov=input('Varients of Covid-19: ')
while .....:
    print('Your input:',.....)
    Cov=input('Varients of Covid-19: ')
```

แบบฝึกหัด

1. จงเติมช่องว่างในโปรแกรมวนรับรหัสนักศึกษาแล้วตรวจสอบตัวเลข 2 ตัวท้ายของรหัส หยุดลูปเมื่อรหัส 2 ตัวท้ายของรหัสเป็น 00 แต่มีเลขท้ายรหัสนักศึกษาเป็น 11 หรือ 22 โปรแกรมจะหยุดการทำงาน โดยกำหนดให้การรับครั้งแรกรหัสนักศึกษาจะต้องไม่ลงท้ายด้วย 11 และ 22

2. จงเติมช่องว่างในโปรแกรมวนรับรหัสนักศึกษาแล้วตรวจสอบตัวเลข 2 ตัวท้ายของรหัส หยุดลูปเมื่อรหัส 2 ตัวท้ายของรหัสเป็น 00 แต่มีเลขท้ายรหัสนักศึกษาเป็น 11 หรือ 22 โปรแกรมจะไม่แสดงเลข 2 ตัวท้าย โดยกำหนดให้การรับครั้งแรกรหัสนักศึกษาจะต้องไม่ลงท้ายด้วย 11 และ 22

PYTHON CODE:

```
n=int(input('ID:'))..........
while n!=00:
    print(n)
    n=int(input('ID:'))..........
    if n==11 or n==22:
        ..........
```

PYTHON CODE:

```
n=int(input('ID:'))..........
while n!=00:
    print(n)
    n=int(input('ID:'))..........
    if n==11 or n==22:
        ..........
```

แบบฝึกหัด

3. จงเติมช่องว่างในโปรแกรมวนรับรหัสนักศึกษาแล้วตรวจสอบตัวเลข 2 ตัวท้ายของรหัส โปรแกรมหยุดการทำงานเมื่อรหัส 2 ตัวท้ายของรหัสเป็น 00 หรือ 11 หรือ 22

PYTHON CODE:

```
while ..... :
    n=int(input('ID:'))%100
    if n==0 ..... n==11..... n==22:
        .....
    else:
        print(n)
```

แบบฝึกหัด

4. จงเขียนโปรแกรมเพื่อวนรับค่ารหัสพนักงานเพื่อแจกเงินพิเศษ หยุดลูปเมื่อรหัสพนักงาน <= 0 แต่มีพนักงานบางคนติด black list (id เป็น 1, 7, 12) ซึ่งเขาจะไม่ได้รับเงินพิเศษ

ตัวอย่างการรันโปรแกรม

Enter id : 5<enter>

5 got extra money

Enter id : 7<enter>

Enter id : 2<enter>

2 got extra money

Enter id : -4<enter>

แบบฝึกหัด

4. จงเขียนโปรแกรมเพื่อวนรับค่ารหัสพนักงานเพื่อแจกเงินพิเศษ หยุดลูปเมื่อรหัสพนักงาน <= 0 แต่มีพนักงานบางคนติด black list (id เป็น 1, 7, 12) ซึ่งเขาจะไม่ได้รับเงินพิเศษ

ถ้าใครยังเขียนเขียนโปรแกรมไม่ได้ ลองนึกคำสั่งได้ประมาณนี้??

Version 1:

```
id=int(input('Enter id : '))
id=int(input('Enter id : '))
print(id, 'got extra money')
id=int(input('Enter id : '))
while id>0:
if id==1 or id==7 or id==12:
continue
```

Version 2:

```
id=int(input('Enter id : '))
print(id, 'got extra money')
while True:
if id>0:
if id==1 or id==7 or id==12:
continue
else:
else:
break
```

ลองเอาชุดคำสั่งข้างบนมาจัดเรียงและย่อหน้าให้เป็นโปรแกรมที่ถูกต้อง

แบบฝึกหัด

5. จงเขียนโปรแกรมวนรับค่ารหัสพนักงานเพื่อแจกเงินพิเศษ หยุดลูปเมื่อรหัสพนักงาน <= 0 แต่มีพนักงานบางคนติด black list (id เป็น 1, 7, 12) ซึ่งเขาจะไม่ได้รับเงินพิเศษ **และทำให้บริษัทหยุดการแจกเงินทันที**

ตัวอย่างการรันโปรแกรม

Enter id : 15<enter>

15 got extra money

Enter id : 6<enter>

6 got extra money

Enter id : 1<enter>

End