

w07-Lec

Iterations

Part III

Assembled for 204111
by Areerat Trongratsameethong
Kittipitch Kuptavanich

Sentinels

- การระบุจำนวนครั้งที่ทำซ้ำไว้เป็นค่าคงที่ในบางครั้งอาจไม่ยืดหยุ่นเพียงพอกับปัญหาที่ต้องการแก้ไข
 - เช่นกรณีต้องการหาค่าเฉลี่ยของคะแนนของนักเรียนในชั้น แต่ไม่ทราบจำนวนนักเรียนล่วงหน้า
- สามารถแก้ปัญหาได้โดย
 - ให้ user ระบุจำนวนครั้งที่ต้องการทำซ้ำ ผ่าน input
 - หรือ อ่าน input จาก user จนเจอค่าที่ตกลงกันไว้ว่าใช้แสดงจุดสิ้นสุดของข้อมูล เช่น -1 หรือ EOF (End of File)
 - ค่าที่ใช้แสดงจุดเริ่มหรือสิ้นสุดของข้อมูลในลักษณะนี้เรียกว่า **sentinels**
 - ควรเลือกค่า sentinels ให้ไม่ทับซ้อนกับค่าของข้อมูลที่เป็นไปได้

Basic Loop Structures

การสร้าง Loop ประกอบด้วย 4 องค์ประกอบหลักคือ

1. Repetition statement: คำสั่งที่ใช้สำหรับการทำซ้ำ
 - ✓ while statement
 - ✓ for statement
2. Condition: เงื่อนไขของการทำซ้ำ
3. A statement that initially sets the condition being tested: การตั้งค่าเริ่มต้นให้กับตัวแปรที่ใช้ในการควบคุมเงื่อนไข
4. A statement within the repeating section of code that alters the condition so that it eventually becomes false: การเปลี่ยนแปลงค่าตัวแปรที่ใช้ในการควบคุมเงื่อนไข ในการวนแต่ละครั้ง เพื่อให้เงื่อนไขเป็นเท็จในที่สุด

Sentinels [2]

```

09 def score_average():
10     SENTINEL = -1
11     print("Please enter students' score one for each line")
12     print("and %d for termination: " % SENTINEL)
13     total = 0.0
14     count = 0
15
16     while (True):
17         score = float(input(""))
18
19         if score == SENTINEL:
20             break
21         total += score
22         count += 1
23
24     if count > 0:          #why do we need this?
25         average = total / count
26     else
27         average = 0
28
29     print("The average of the %d numbers is %8.4f" %(count, average))

```

Nested Loop

Pseudocode

Initial Condition: row = 1

WHILE row <= 10

Initial Condition: col = 1

WHILE col <= 10

Display col

Add 1 to col

ENDWHILE

Add 1 to row

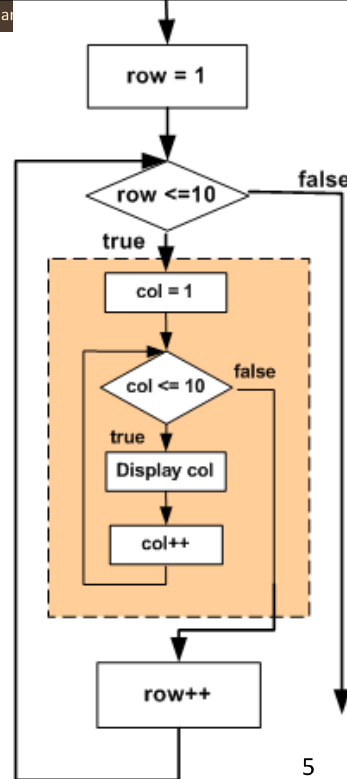
ENDWHILE

Inner Loop

Outer Loop

Reference:

<http://facweb.northseattle.edu/voffenba/class/CSC110-W08/NotesDL/wk08/NoteOnNestedLoops.htm>



5

Nested Loop [2]

```
02 def nestedLoop():
```

```
03
```

```
04     ROW = 5
```

```
05     COLUMN = 4
```

```
06
```

```
07     for i in range(1, ROW + 1):
```

//loop วน starts

```
08         print("\ni is now %d" % i)
```

```
09
```

```
10         { for j in range(1, COLUMN + 1):
```

//loop วน starts

```
11             print("j = %d" % j, end=" ")
```

```
i is now 1
j = 1 j = 2 j = 3 j = 4
i is now 2
j = 1 j = 2 j = 3 j = 4
i is now 3
j = 1 j = 2 j = 3 j = 4
i is now 4
j = 1 j = 2 j = 3 j = 4
i is now 5
j = 1 j = 2 j = 3 j = 4
```

6

Nested Loops (3)

```
i = 1
j = 1 j = 2 j = 3 j = 4
i = 2
j = 1 j = 2 j = 3 j = 4
i = 3
j = 1 j = 2 j = 3 j = 4
i = 4
j = 1 j = 2 j = 3 j = 4
i = 5
j = 1 j = 2 j = 3 j = 4
```

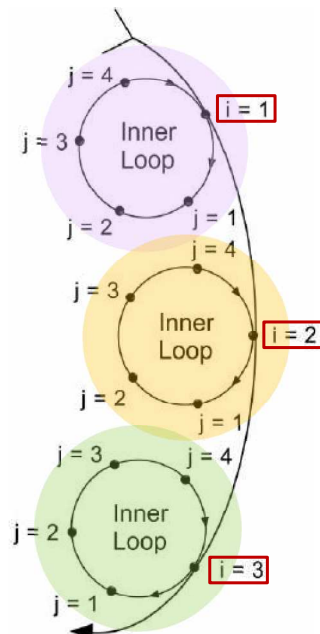


Figure 5.9 j loops once for each i

7

Example 1

```
08 SIZE = 5
```

```
09
```

```
10 for i in range(SIZE):
```

#outer loop starts

```
11     for j in range(i + 1):
```

#inner loop starts

```
12         print(j + 1, end=" ")
```

```
13
```

#inner loop ends

```
14     print("")
```

```
15
```

#outer loop end

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

iteration	i	j	output
1	0	0	1
2	1	0 ... 1	1 2
3	2	0 ... 2	1 2 3
4			
5			

8

Example 2

```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

```
08 SIZE = 5
09
10 for i in range(SIZE):           #outer loop starts
11     for j in range(SIZE - i):   #inner loop starts
12         print(j + 1, end=" ")
13                                 #inner loop ends
14     print("")
15                                 #outer loop end
```

iteration	i	j	output
1	0	0 ... 4	_____
2	1	_____	_____
3	2	_____	_____
4	3	_____	_____
5	4	_____	_____

9

Practice 1

ให้เขียนฟังก์ชัน `rectagle1(n)` $n (n \geq 1)$ และแสดงผลดัง output

```
Enter n: 5
1 2 3 4 5
 2 3 4 5
   3 4 5
    4 5
     5
```

- ตัวอย่างการ run 2

```
Enter n: 3
1 2 3
 2 3
  3
```

11

Example 3

```
  *
 * *
* * *
* * * *
* * * * *
```

```
08 SIZE = 5
09
10 for i in range(SIZE):
11     for j in range(SIZE - i):
12         print(" ", end="")
13
14     for k in range(i + 1):
15         print("* ", end="")
16
17     print("")
```

iteration	i	j	k	output
1	0	_____	_____	_____
2	1	_____	_____	_____
3	2	_____	_____	_____
4	3	_____	_____	_____
5	4	_____	_____	_____

10

Practice 2

ให้เขียนฟังก์ชัน `rectagle2(n)` $n (n \geq 1)$ และแสดงผลดัง output

```
Enter n: 5
      5
     4 5
    3 4 5
   2 3 4 5
  1 2 3 4 5
```

- ตัวอย่างการ run 2

```
Enter n: 3
      3
     2 3
    1 2 3
```

12

Practice 3

ให้เขียนฟังก์ชัน `rectagle3(n)` $n (n \geq 1)$ และแสดงผลดัง output

```
Enter n: 5
5 4 3 2 1
 4 3 2 1
   3 2 1
    2 1
     1
```

- ตัวอย่างการ run 2

```
Enter n: 3
3 2 1
 2 1
  1
```

13

Practice 5

ให้เขียนโปรแกรมเพื่อรับค่าจำนวนเต็ม (-, 0, +) และแสดงค่าจำนวนที่กลับหลักแล้ว ดังผลการ run ด้านล่าง

- ตัวอย่างการ run 1

```
Input an integer: 94857
75849
```

- ตัวอย่างการ run 2

```
Input an integer: -1
-1
```

15

Practice 4

ให้เขียนโปรแกรมเพื่อรับค่าจำนวนเต็มที่ประกอบด้วย 1 และ 0 ไม่เกิน 8 หลัก พร้อมแสดงผลให้อยู่ในรูป เลขฐาน 2 จำนวน 8 bit โดยให้ใส่ 0 ในจำนวนหลักที่ว่าง

- ตัวอย่างการ run 1

```
Input binary: 1101
|0000 1101| is 13 in decimal
```

- ตัวอย่างการ run 2

```
Input binary: 1102
Invalid input
Input binary: 11
|0000 0011| is 3 in decimal
```

14

Practice 6

ให้เขียนโปรแกรมเพื่อรับเลขจำนวนเต็ม x แล้วตรวจสอบว่า x เป็นจำนวนเฉพาะหรือไม่ ดังตัวอย่างการ run ด้านล่าง

- ตัวอย่างการ run 1

```
Input an integer: 1223
1223 is prime
```

- ตัวอย่างการ run 2

```
Input an integer: 1227
1227 is NOT prime
```

หมายเหตุ: จำนวนแรกที่ยก x ลงตัวหาก x ไม่ใช่จำนวนเฉพาะจะมีค่าไม่เกิน \sqrt{x}

16

Practice 7

ให้เขียนโปรแกรมเพื่อคำนวณเลข fraction ในฐาน 2 โดยรับค่าเฉพาะส่วนที่อยู่หลังจุด binary point ไม่เกิน 8 ตำแหน่ง พร้อมแสดงผลเป็นทศนิยม 8 ตำแหน่ง ดังผลการ run ด้านล่าง

```
Input binary fraction: 1101
0.1101 is the sum of
1/2 +
1/4 +
0/8 +
1/16
The result is 0.81250000
```

Practice 8

ให้นำเข้าข้อมูลประเภทจำนวนเต็ม 2 ค่า คือค่าแรก (first) และค่าสุดท้าย (last) ผ่านทาง keyboard และนับว่ามีจำนวนเฉพาะกี่จำนวน อะไรบ้าง

- ตัวอย่างการ run

```
Input first integer: 3
Input last integer: 20
Prime numbers between 3 and 20 are:
3 5 7 11 13 17 19
Total: 7
```