

## Description

ถ้าหากทางเลือกเพียงสองทางยังไม่จูงใจคุณ

ฉันขอเสนอการสร้างเงื่อนไขลูกโซ่(Chained) และ เงื่อนไขซ้อน(Nested) สามารถใช้งานในสถานการณ์ที่โปรแกรมต้องการทางเลือกหลายรูปแบบ เมื่อเงื่อนไขเพียงหนึ่ง(if) หรือสองทาง(if-else) ไม่สามารถตอบโจทย์คุณได้

ถ้าฉันทำรูป มันอาจจะใหญ่เกินกว่าจะเป็นเพียงโจทย์ปัญหา  
เอาเป็นว่าคุณอ่าน Think Python ประกอบไปด้วยละกัน

---

Chained Condition คือการสร้างเงื่อนไขในลักษณะเป็นลูกโซ่ตามชื่อ  
เป็นการสร้างเงื่อนไขเพิ่มเติม หลังจากเงื่อนไขก่อนหน้าเป็นเท็จ  
เช่นถ้ามี if statement ที่หนึ่ง  
แทนที่คุณจะใช้ else เพื่อทำงานชุดคำสั่งทางเลือก

คุณอาจจะต้องการเงื่อนไขที่สอง หลังจากเงื่อนไขที่หนึ่งเป็นเท็จมา  
และคุณอาจจะต้องการเงื่อนไขที่ สาม, สี่, ห้า, ห้า+1, ห้า+2, N  
คุณก็เขียน statement ในลักษณะเดียวกันดังนี้

```
"""
For education propose ~
"""

def core_function(option_number):
    """
    Do something that i don't know too.
    """
    if option_number == 1:
        function_option_1()
    elif option_number == 2:
        result = function_option_2()
        return result
    elif option_number == 3:
        pass
    elif option_number == 4:
        function_option_4()
        function_option_4()
        function_option_4()
        function_option_4()
        function_option_4()
        function_option_4()
        function_option_4()
    else:
        print("Nothing happens.")
```

**\*\*elif เป็นการรวมกันระหว่าง else กับ if ชนิดจับคู่มาชนกันดี้อะ**  
**\*\*เพราะฉะนั้นจะมี elif ได้ ต้องมี if เริ่มต้นเงื่อนไขแรกก่อนเสมอ**  
**\*\*ฉันเขียนเรื่องอื่นแทรกลงไปด้วย ฉันจะอธิบายสั้นๆ**

ถ้าหาก option\_number == 1 เป็นเท็จในเงื่อนไขแรก  
ตัวโปรแกรมจะลงมาดู elif <เงื่อนไข> ถัดไป  
การทำงานก็จะเหมือนกับ if ทั่วไป ไม่ได้มีอะไรต่างกัน

การที่ Python มี elif เป็นเพราะว่า

โปรแกรมใดๆในโลกล้วน มีการใช้งาน Condition ในลักษณะของ Chained Conditional  
และเมื่อประกอบกับ Python ใช้ Indentation ในการครอบคลุมเพื่อสร้าง Compound Statement  
ถ้าไม่มี elif โปรแกรมของคุณจะมีการใช้ Indent ที่สลับซับซ้อนมาก ชุดคำสั่งของคุณจะถูกเขียนโดยมีที่หน้าตา  
เหมือนบันได ซึ่งไม่ได้มีความสวยงามเลย

pass statement คือคำสั่งที่... "ไม่ได้ทำอะไรทั้งนั้น" ประมาณว่า "ไม่มีอะไร ช่างฉันเถอะ"  
มีไว้เพื่อ"อุด"โปรแกรมบางส่วนที่คุณเห็นว่า ไม่ได้จะทำอะไรทั้งนั้นภายใน Compound statement นั้นๆ  
แต่ Python มีข้อกำหนดว่าจะปล่อย Indent ช่วงนั้นให้ว่างโดยที่ไม่มีคำสั่งอะไรไม่ได้ (ใช้ Comment # ก็ไม่ได้)  
แต่คุณไม่มีคำสั่งอะไรที่จะให้ทำงานจริงๆ คุณสามารถใช้คำสั่ง pass ได้  
แต่ถ้าคุณใช้ pass โดยที่มีคำสั่งอื่นด้วย ถือว่าเป็นความสลับซับซ้อนอย่างหนึ่ง

ต่อไปเป็น Nested Condition

ในบางครั้งคุณอาจต้องการสร้างเงื่อนไขซ้อนขึ้นมาภายในเงื่อนไขจริงๆ โดยที่ elif ไม่สามารถตอบโจทย์คุณได้  
ในภาษา Python นั้นไม่ได้มีรูปแบบที่พิเศษหรือประหลาดไปกว่าภาษาอื่น  
คุณสามารถเขียนเงื่อนไขในลักษณะของ Nested ได้ดังนี้

```
"""
For education propose ~
"""

def core_function(option_number):
    """
    Do something that i don't know too.
    """
    if option_number == 1:
        sub_option = function_option_1()
        if sub_option == 1:
            sub_process_1()
        if sub_option == 2:
            print("No!!!")
            return 1
        if sub_option == 3:
            sub_sub_option = do_something()
            if sub_sub_option:
                pass
            else:
                print("Yes~")
                return 0
        print("LINE")
    else:
        print("1234567890")
        return 10
    print("Why are you here?")
```

\*\*เขียนลงไปดี๊อย่างนั้นเลยนั่นแหละ

ให้สังเกต if-else ภายในส่วนของ if แรกสุด

เขียนให้เหมือนเป็นอีกคำสั่งหนึ่งทั้งอย่างนั้นแหละ ฉันก็ไม่ว่าจะอธิบายด้วยคำยั้งใจ  
เกรงว่าโจทย์จะยาวไปกว่านี้

การเขียนโปรแกรมเชิงหัด Chained และ Nested สามารถใช้สร้างเงื่อนไขได้ทั้งหมด

การเขียนโปรแกรมของ C++ และ Java สามารถเขียนร่วมกันได้ทั้งหมด

และคุณอาจจะต้องใช้อีกบ่อยครั้ง ขอให้คุณดูความเหมาะสมของปัญหา และตัวโปรแกรมของคุณเอง ว่าคุณ  
อยากจะเขียนอย่างไร

งานของคุณคือ ให้รับข้อมูลทางเลือกมาก่อนว่าเป็น Ascend หรือ Descend  
และรับตัวเลขมาสามตัว

ถ้าทางเลือกเป็น Ascend

ให้แสดงผลตัวเลขทั้งสามตัว จากน้อยไปมาก

ถ้าทางเลือกเป็น Descend

ให้แสดงผลตัวเลขทั้งสามตัว จากมากไปน้อย

แสดงผลตัวเลขทั้งสามตัว อยู่ในบรรทัดเดียวกัน มีลูกน้ำ(comma)และเว้นวรรค(space)คั่น หากสงสัยให้ดู  
Sample Output

by นายพิชาร เอกอุ่น (<https://ejudge.it.kmitl.ac.th/account/7>)

13 September 2016, 12:45

## Specification

### Input Specification

4 บรรทัด

LINE 1: option [Ascend or Descend]

LINE 2-4: 3 numbers (Real numbers)

### Output Specification

บรรทัดเดียว

ตัวเลขสามตัวแสดงเป็นทศนิยมสองตำแหน่ง เว้น  
วรรคตามที่กำหนด

## Sample Case

### Sample Input

Ascend

5

4

6

Descend

546.78

14.247

744.5122

### Sample Output

4.00, 5.00, 6.00

744.51, 546.78, 14.25

## Time Remaining

0

Day

0

Hour

56

Minutes

49

Seconds

82

## Information

**Time Limit**

1 Second

**Memory Limit**

32 MB

**Language**

py

**Deadline**

21 September 2016, 21:00

**Submission Limit**

Unlimit

**Restrict Word**

for while min max sort

**Testcase**

14 case

**Rejudge Testcase**

0 case

**Full Score**

100 Point

**Bonus Score**

100

**Your Score**

PPPPPPPPPPPPPP

 (<https://ejudge.it.kmitl.ac.th/problem/submission/321783>)

## Submit File

Choose File

 No file chosen

Submit