

Description

โจทย์ปัญหาข้อนี้จะเสนอวิธีการเขียนโปรแกรมให้มีการทำงานซ้ำ (Iteration)

ในหลายๆครั้งคุณต้องสั่งให้โปรแกรมทำงานบางอย่างซ้ำๆจำเจ อย่างที่คุณได้ทำมาในโจทย์ข้อที่ผ่านๆมา การเขียนโปรแกรมให้มีการทำงานซ้ำมีความสำคัญอย่างมากในโปรแกรมคอมพิวเตอร์ เพราะจะสามารถช่วยลดขนาดของโปรแกรมลง ทั้งยังเพิ่มความยืดหยุ่น และความหลากหลาย รวมถึงความสามารถของโปรแกรม

สมมติว่าคุณถูกคุณครูลงโทษ เนื่องจากคุณเล่นเครื่องร่อนกระดาษในห้องเรียน คุณครูต้องการสั่งให้คุณคิดข้อความว่า "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน" 10 จบ คุณครูจะสั่งคุณว่าอย่างไร ระหว่าง

A)

เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"
 เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน"

... เหนื่อยแย่เลย

หรือ

B)

เธอไปคิดข้อความ "ฉันจะไม่ขว้างเครื่องร่อนกระดาษในห้องเรียน" 10 จบเดี๋ยวนี้

ง่ายกว่ากันเยอะ

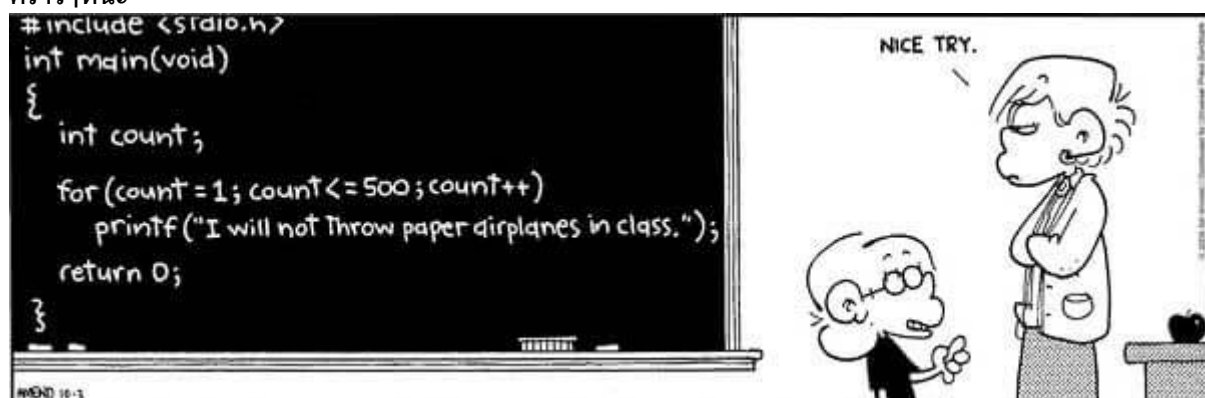
แถมแบบ **B)** คุณครูยังสามารถเปลี่ยนจำนวนครั้งที่ให้คุณคิดได้อีกด้วย โดยที่ไม่ต้องสั่งซ้ำซาก เช่นถ้าเปลี่ยนเป็น 100 จบ หรือ 500 จบ คุณครูก็เพียงเปลี่ยนจาก 10 เป็น 500

ปล.แต่คุณที่เป็นคนคิด ก็ยังต้องคิดในปริมาณตามที่สั่งอยู่ดี ไม่ว่าจะใช้วิธี **A)** หรือ **B)**

เปรียบเสมือนคุณครูเป็นโปรแกรม และคุณเป็นคอมพิวเตอร์

คุณครูมีวิธีสั่งให้คุณทำตามมากมาย แต่ไม่ว่าจะใช้วิธีใด ท้ายสุดคุณยังคงต้องทำงานแบบเดิมอยู่ดี

คร่าวๆนะ



(อันนี้การดู... บนกระดานเป็นภาษา C ฉันใส่มาประกอบเพื่อให้ดูมีอะไรเฉยๆ)

ในภาษา Python

การเขียนโปรแกรมที่มีการทำงานซ้ำนั้น คุณสามารถใช้ Statements ได้สองอย่างคือ for และ while
ฉันจะเสนอ for ก่อน เพราะว่าเข้าใจง่ายกว่าเล็กน้อย

```
for count in range(500):  
    print("I will not throw paper airplanes in class.")
```

for statement นั้นเขียนด้วย keyword for __variable__ in __iterable__
for กับ in เป็น keyword ต้องใส่เสมอ

__iterable__ ที่ใช้สำหรับ for statement นั้น เป็น object อะไรก็ได้ที่สามารถ iterable ได้
iterable ในที่นี้หมายความว่าสามารถทำซ้ำได้ โดยมีลำดับกำหนดชัดเจน
คุณอ่านเฉยๆน่าจะเข้าใจ ฉันยกตัวอย่างกันใช้งาน built-in function range
range นั้นสามารถใช้ได้ง่าย เมื่อต้องการลำดับเลขคณิตอย่างง่าย เช่น 0 ถึง 9; 1 ถึง 99 หรือ 100 ถึง 9999
เพิ่มขึ้นทีละ 3
range ถูกออกแบบมาเป็น function ที่มี optional parameter ที่คุณสามารถกำหนดได้ถึงสองตัว คือ stop และ
step
คุณสามารถศึกษาทดลองวิธีใช้งานได้ด้วยตัวเอง

โดยที่ range จะ return iterable object (object range) มาให้ ซึ่งสามารถทำงานร่วมกับ for ได้

ในตัวอย่างนี้เป็น object ที่ได้จาก range(500)
object นี้จะมีลำดับทั้งหมด 500 ลำดับ
โดยที่แต่ละลำดับมีค่า(value) ตั้งแต่ 0 ถึง 499
[0, 1, 2, 3, ..., 498, 499]

ซึ่ง for statement นี้จะไล่ลำดับตั้งแต่ลำดับแรก จนถึงลำดับสุดท้าย
แล้วทำงานตามชุดคำสั่งที่เขียนลงไป ในลักษณะ compound statement อย่างที่คุณเคยเขียนกันมา
เกิดเป็นการทำงานซ้ำๆกันภายใน for statement นี้

และ __variable__ เป็นตัวแปรที่ระบุ หรือกำกับค่าของลำดับ ในแต่ละลำดับที่ for loop นั้นๆทำงาน
(กำกับค่าของลำดับ ไม่ใช่กำกับลำดับ แต่พอดีถ้าคุณใช้ range(500); ลำดับ และค่าของลำดับมันจะเหมือนกัน
พอดี)
ซึ่งแต่ละรอบนั้น ค่าของ __variable__ จะเปลี่ยนไปเรื่อยๆ
คุณสามารถใช้ประโยชน์จากการเปลี่ยนแปลงของค่า __variable__ ในแต่ละรอบได้
หรือไม่ใช้เลยก็ได้
แต่ __variable__ ที่ใช้กำกับค่าลำดับนั้น (ปกติฉันจะเรียกว่า ตัวแปรที่ใช้วิ่งใน loop ซึ่งกำกวม เลยไม่ได้เรียก
บ่อยๆ)
คุณควรใช้ด้วยความระมัดระวัง การเปลี่ยนแปลงค่าใดๆของ __variable__ ภายในรอบนั้นๆ จะไม่ส่งผลกระทบต่อ
รอบถัดไปของ loop

และค่าที่ใช้เป็น __iterable__; คุณไม่ควรแก้ไข หรือเปลี่ยนแปลงในระหว่าง loop

ฉันรู้สึกว่ามันยาวเกินไปแล้ว ตัดจบแค่นี้ก่อนละกัน

งานของคุณคือ ให้รับข้อความ w
ให้แสดงผลข้อความ w ซ้ำๆ จำนวน 100 รอบ

Specification

➡ Input Specification

หนึ่งบรรทัด
w เป็นข้อความ

➡ Output Specification

100 บรรทัด
เป็นข้อความ w ในแต่ละบรรทัด บรรทัดละ 1 ข้อความ

Sample Case

➡ Sample Input

Hello world!

➡ Sample Output

```
Hello world! # <- บรรทัดที่ 1
Hello world!
Hello world!
...
... # <- จัณยขอให้ เดี่ยวว่ามันจะยาวไป
...
Hello world! # <- บรรทัดที่ 100
```

Time Remaining

4

Days

3

Hours

12

Minutes

57

Seconds

616

Information

Time Limit

1 Second

Memory Limit

32 MB

Language

py

Deadline

5 December 2017, 23:59

Submission Limit

Unlimit

Restrict Word