

## ผังเมือง

0.33 seconds, 32 megabytes

\*\*\* อนุญาตให้ใช้ sort() จาก <algorithms> \*\*\*

เกรซเป็นนักวางผังเมืองที่ต้องการจะพัฒนาเมืองโดยที่เมืองนี้มีบ้าน  $V$  หลังและมีถนนแบบทางเดียว (one-way) จำนวน  $E$  เส้นเกรซต้องการที่จะให้เมืองนี้สามารถขับรถไปและกลับมาที่จุดเดิมได้

เกรซจะจัดบ้านสองหลังให้อยู่ในหมู่บ้านเดียวกัน หากสามารถเดินทางจากบ้านหลังหนึ่งไปยังอีกหลังหนึ่งได้และสามารถเดินทางกลับมายังบ้านหลังเดิมได้ จงเขียนโปรแกรมเพื่อช่วยเกรซหาขนาดหมู่บ้านที่ใหญ่ที่สุดในเมือง 5 อันดับแรกโดยเรียงจากมากไปน้อยหากในเมืองมีจำนวนหมู่บ้านไม่ถึง 5 แห่งให้เติม 0 จนกว่าจะครบ 5 จำนวนโดยให้แสดงในรูปแบบที่ใช้ "," คั่นระหว่างตัวเลข เช่น "5,4,3,2,1" โดยที่คำตอบต้องไม่มี " " และถ้าไม่สามารถแสดงตัวเลขได้ครบ 5 ตัวได้ ให้เติม 0 เข้าไปจนกว่าจะครบเลข 5 ตัว เช่น "5,4,3,0,0"

บ้านที่สามารถไปบ้านหลังอื่นได้แต่ไม่สามารถกลับมาหลังตัวเองได้ให้นับเป็นหมู่บ้านที่มีขนาดเท่ากับ 1

## ข้อมูลนำเข้า (Input)

บรรทัดแรก      ระบุจำนวน  $V$  และ  $E$  โดยที่  $V, E \in \mathbb{Z}^+$  และ  $2 \leq V \leq 50,000$   
                          และ  $E \leq 100,000$   
 บรรทัดที่  $|E| + 1$     บ้านหลังที่  $U$  กับ  $V$  โดยที่บ้าน  $U$  มีถนนทางเดียวไปที่บ้าน  $V$

## ข้อมูลนำออก (Output)

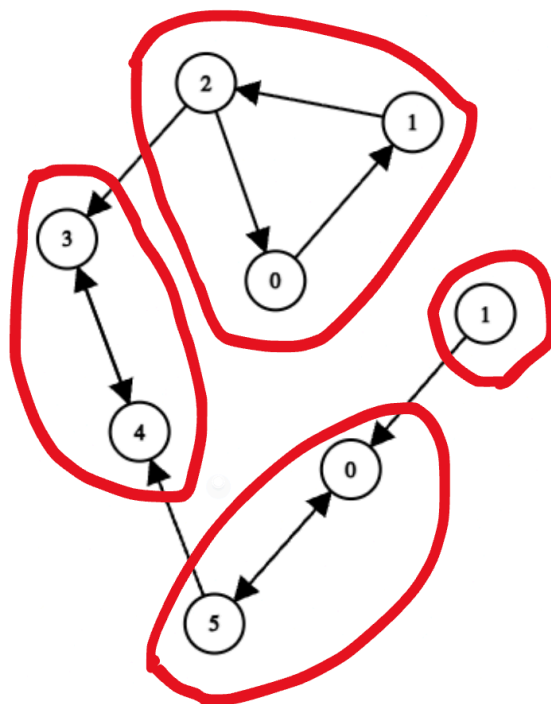
บรรทัดแรก    ให้แสดงเลข 5 ตัวจากมากไปน้อยที่เป็นขนาดของกลุ่มบ้านที่สามารถไปกลับได้

## ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออก (Input/Output Examples)

ข้อมูลนำเข้า	ข้อมูลส่งออก
8 10 1 2 2 3 3 1 3 4 4 5 5 4 6 5 6 7 7 6 8 7	3, 2, 2, 1, 0

ข้อมูลนำเข้า	ข้อมูลส่งออก
7 10	1, 1, 1, 1, 1
4 5	
2 5	
3 4	
2 4	
1 5	
1 2	
3 5	
7 3	
5 6	
2 7	

จำนวนชุดทดสอบ: 11 ชุด



รูปภาพกราฟตัวอย่างของ input และ output ชุดที่ 1

Hint :

1. [Method 1](#)
2. [Method 2](#)

Extra :

If your biggest output is 1, it means the graph has a topological order: a directed graph has a topological order iff all SCCs have size 1.

So I want to know how many topological orders the DAG has. this changes the problem statement from  $P$  to  $\#P$ .  $P$  and  $\#P$  are computational complexity classes, which describe how efficiently we can solve a problem as the input grows.

So if you are interested in what computational complexity classes are, I recommend this:

1. [what is topological order](#)
2. [what is computational complexity classes](#)
3. [what is #P](#)
4. Computers and Intractability: A Guide to the Theory of NP-Completeness by Michael R. Garey and David S. Johnson
5. Introduction to the Theory of Computation by Michael Sipser
6. Computational Complexity a Modern Approach by Sanjeev Arora and Boaz Barak