

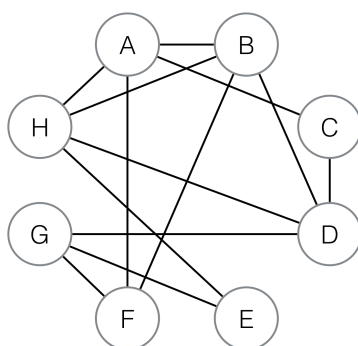
# Friend Suggestion

memory limit: 32 MB

time limit: 1 s

เครือข่ายสังคม (social network) ต่าง ๆ ในปัจจุบัน จะมีความสามารถในการแนะนำสมาชิกที่ผู้ใช้งานน่าจะรู้จัก เรียกว่า *friend suggestion* ในโจทย์ข้อนี้ เราจะมาศึกษาวิธีพื้นฐานหนึ่งในการทำ friend suggestion โดยพิจารณาจากสมาชิกที่ผู้ใช้งานเป็นเพื่อนกันในระบบอยู่แล้ว

สมมติว่าเครือข่ายสังคมแห่งหนึ่ง มีสมาชิก 8 คนได้แก่ A, B, C, D, E, F, G, และ H และความสัมพันธ์การเป็นเพื่อนกันในระบบแสดงในกราฟด้านล่าง (สมาชิกที่มีเส้นเชื่อมถึงกันจะเป็นเพื่อนกันอยู่แล้วในระบบ)



พิจารณาผู้ใช้งาน A จะเห็นว่าในระบบนั้น A มีเพื่อนอยู่ 4 คนได้แก่ B, C, F, และ H สมาชิกในระบบที่ไม่ได้เป็นเพื่อนกับ A ได้แก่ D, E, และ G

หากเรานับสมาชิกที่เป็นเพื่อนของทั้ง A และ D จะมีอยู่ 3 คนได้แก่ B, C, และ H เรากล่าวว่า A และ D มี *เพื่อนร่วมกัน* (mutual friend) 3 คน เมื่อพิจารณาในทำนองเดียวกัน จะได้ว่า A และ E มีเพื่อนร่วมกัน 1 คน, A และ G มีเพื่อนร่วมกัน 1 คน

หากเราตั้งสมมติฐานว่าสมาชิกที่มีเพื่อนร่วมกันมาก น่าจะรู้จักกัน จะสรุปได้ว่า ในบรรดาสมาชิกที่ A ไม่รู้จัก เขาน่าจะรู้จัก D มากที่สุด เราสามารถใช้ข้อสรุปนี้ในการแนะนำ D, E, และ G ให้ A ได้ โดยจัดอันดับให้ D เป็นอันดับแรก (เพราะ A น่าจะรู้จัก D มากกว่า E และ G)

ในโจทย์ข้อนี้ เราต้องการให้คุณเขียนโปรแกรมแนะนำสมาชิกที่ผู้ใช้งานหนึ่งน่าจะรู้จักตามลำดับ เมื่อได้รับข้อมูลเป็นสถานะความเป็นเพื่อนของสมาชิกทั้งหมดในเครือข่ายสังคม

## Input

ข้อมูลเข้าบรรทัดแรกเป็นจำนวนเต็ม  $n$  และ  $m$  แทนจำนวนสมาชิกในเครือข่ายสังคม และจำนวนคู่ของสมาชิกที่เป็นเพื่อนกันในระบบ ( $1 < n \leq 1000$ ,  $0 \leq m < \binom{n}{2}$ )  $m$  บรรทัดถัดมา แต่ละบรรทัดเป็นจำนวนเต็ม  $i$  และ  $j$  โดย  $1 \leq i, j \leq n$  แทนหมายเลขของสมาชิกสองคนที่เป็นเพื่อนกันในระบบ บรรทัดต่อมาเป็นจำนวนเต็ม  $k$  ที่  $1 \leq k \leq n$  แทนหมายเลขสมาชิกที่ต้องการให้คุณแนะนำเพื่อนให้

## Output

ให้โปรแกรมแสดงคู่ของจำนวนเต็ม  $i$  และ  $f$  ต่อหนึ่งบรรทัด โดย  $i$  แทนหมายเลขสมาชิกที่ผู้ใช้หมายเลข  $k$  ยังไม่รู้จัก และ  $f$  แทนจำนวนเพื่อนร่วมกันระหว่าง  $i$  และ  $k$  ตามลำดับ ให้โปรแกรมแสดงรายการดังกล่าวจำนวน 10 บรรทัด โดยเรียงให้สมาชิกที่จำนวนเพื่อนร่วมกันมากที่สุดขึ้นก่อน หากมีสมาชิกหลายคนที่มีจำนวนเพื่อนร่วมกันกับ  $k$  เท่ากัน ให้แสดงสมาชิกที่มีหมายเลขน้อยขึ้นก่อน

หากสมาชิกที่  $k$  ยังไม่รู้จักมีน้อยกว่า 10 คน ให้แสดงทั้งหมดตามลำดับดังกล่าว

## Example

### Input

```
8 13
2 1
1 6
6 7
1 8
2 4
6 2
3 4
4 7
1 3
4 8
5 7
2 8
8 5
1
```

### Output

```
4 3
5 1
7 1
```