

Kerékpártúra

Áron szeret kerékpározni erdei útvonalakon. A hétvégére kinézett magának egy olyan erdei helyszínt, ahol eddig még nem járt. Az erdei helyszíneket és az őket összekötő közvetlen kerékpárutakat csatlakozó pontokkal, valamint azokat összekötő szakaszokkal azonosítjuk. Áronnak van térképe a terepről, ami pontosan megadja, hogy mely csatlakozó pontokat köt össze szakasz. Ezek a szakaszok általában egyirányúak, de lehet olyan is, amelyik mindkét irányban járható.

Már kijelölte a K kezdőpontot. Mivel nem ismeri a terepet, ezért célpontnak olyat akar választani, amelyhez vezet olyan útvonal, hogy ha nem bírja tovább folytatni, akkor bármely, az útvonalba eső pontból (a célpont kivételével) vissza tudjon menni a kiindulási pontba.

Készíts programot, amely megadja az összes lehetséges célpontot!

Bemenet

A *standard bemenet* első sorában a csatlakozó pontok száma ($1 \leq N \leq 10\,000$), a szakaszok száma ($1 \leq M \leq 200\,000$), és a kezdőpont ($1 \leq K \leq N$) van.

A következő M sor mindegyike rendre egy szakasz végpontjait tartalmazza ($1 \leq U \neq V \leq N$), ami azt jelenti, hogy U -ból V -be lehet haladni a szakaszon. A csatlakozó pontokat az $1, \dots, N$ számokkal azonosítjuk.

Kimenet

A *standard kimenet* első sorába a lehetséges célpontok C számát kell írni! A második sor pontosan C egész számot tartalmazzon, a lehetséges célpontok azonosítóit, tetszőleges sorrendben!

Ha nincs a feltételeknek megfelelő célpont, akkor az első és egyetlen sor a 0 számot tartalmazzza!

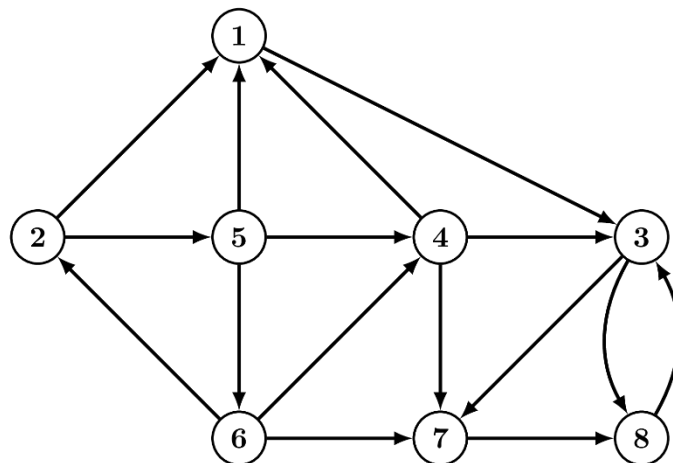
Példa

Bemenet

```
8 16 2
1 3
2 1
2 5
5 4
5 1
5 6
4 3
4 7
4 1
3 8
3 7
6 2
6 4
6 7
7 8
8 3
```

Kimenet

```
5
5 4 6 7 1
```



Korlátok

Időlimit: 0.4 mp.

Memórialimit: 32 MB

Pontozás

A pontok 24%-a szerezhető olyan tesztekre, ahol $N \leq 100$.

A pontok további 24%-a szerezhető olyan tesztekre, ahol $N \leq 1000$.