

Game

Օ-ից $n - 1$ թվերով համարակալված n մոլորակ հայտնաբերելուց հետո փարավոնները սկսեցին նրանց միջև կառուցել տրանսպորտային համակարգ **մեկ-կողմանի տեղեպորտերների** միջոցով: Յուրաքանչյուր տեղեպորտեր ունի սկզբնական մոլորակ և վերջնական մոլորակ: Երբ տուրիստն օգտագործում է տեղեպորտերը սկզբնական մոլորակում նա տեղափոխվում է վերջնական մոլորակ: Նկատենք, որ տեղեպորտերի սկզբնական և վերջնական մոլորակները կարող են նույնը լինել: Սկզբնական u մոլորակից վերջնական v մոլորակ տեղափոխող տեղեպորտերը կնշանակենք (u, v) :

Տեղեպորտային համակարգի լայնածավալ օգտագործումը խրախուսելու համար փարավոնները ստեղծել են մի խաղ, որը կարող են խաղալ զբոսաշրջիկները տրանսպորտային համակարգով ճանապարհորդելիս:

Տուրիստը կարող է խաղը սկսել ցանկացած մոլորակում: $0, 1, \dots, k - 1$ ($k \leq n$) մոլորակները կոչվում են **հատուկ մոլորակներ**: Ամեն անգամ, երբ տուրիստը մտնում է հատուկ մոլորակ, նա ստանում է կնիք:

Ներկա պահին, յուրաքանչյուր i ($0 \leq i \leq k - 2$) համար գոյություն ունի $(i, i + 1)$ տեղեպորտեր: Այս $k - 1$ տեղեպորտերները կոչվում են **սկզբնական տեղեպորտերներ**:

Նոր տեղեպորտերներն ավելանում են մեկ առ մեկ: Նոր տեղեպորտերներ ավելացնելուց հնարավոր է, որ տուրիստն ունենա անվերջ քանակով կնիքներ ստանալու հնարավորություն: Ավելի ճշգրիտ, դա տեղի է ունենում, երբ գոյություն ունի մոլորակների $w[0], w[1], \dots, w[t]$ հաջորդականություն, որը բավարարում է հետևյալ պայմաններին.

- $1 \leq t$
- $0 \leq w[0] \leq k - 1$
- $w[t] = w[0]$
- Յուրաքանչյուր i ($0 \leq i \leq t - 1$) համար գոյություն ունի $(w[i], w[i + 1])$ տեղեպորտեր:

Նկատենք, որ տուրիստը կարող է օգտագործել սկզբնական տեղեպորտերները և մինչև տվյալ պահին ավելացրած **բոլոր** տեղեպորտերները:

Ձեր խնդիրն է օգնել փարավոններին յուրաքանչյուր տեղեպորտեր ավելացնելուց հետո ստուգել, արդյոք տուրիստը կարող է անվերջ քանակությամբ կնիքներ ստանալ, թե ոչ:

Իրականացման մանրամասներ

Դուք պետք է իրականացնեք հետևյալ ենթածրագրերը.

```
init(int n, int k)
```

- n : մոլորակների քանակը:
- k : հատուկ մոլորակների քանակը:
- Այս ենթածրագիրը կանչվում է ճիշտ մեկ անգամ նախքան `add_teleporter`-ի կանչերը:

```
int add_teleporter(int u, int v)
```

- u և v : ավելացվող տեղեկորտերի սկզբնական և վերջնական մոլորակները:
- Այս ֆունկցիան կանչվում է առավելագույնը m անգամ (m -ի արժեքի մասին տե՛ս Սահմանափակումներում):
- Այս ֆունկցիան պետք է վերադարձնի 1- եթե (u, v) տեղեկորտերն ավելացնելուց հետո տուրիստը կարող է անվերջ քանակությամբ կնիքներ ստանալ: Հակառակ դեպքում ֆունկցիան պետք է վերադարձնի 0:
- Հենց որ այս ֆունկցիան վերադարձնի 1, ձեր ծրագրի աշխատանքը կընդհատվի:

Օրինակներ

Օրինակ 1

Դիտարկենք հետևյալ կանչը.

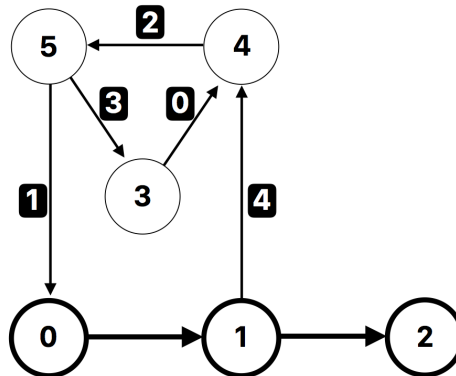
```
init(6, 3)
```

Այս օրինակում կա 6 մոլորակ և 3 հատուկ մոլորակ: 0, 1 և 2 մոլորակները հատուկ մոլորակներ են: (0,1)-ը և (1,2)-ը սկզբնական տեղեկորտերներ են:

Ենթադրենք, գրեյդերն անում է հետևյալ կանչերը.

- (0) `add_teleporter(3, 4)`. Դուք պետք է վերադարձնեք 0:
- (1) `add_teleporter(5, 0)`. Դուք պետք է վերադարձնեք 0:
- (2) `add_teleporter(4, 5)`. Դուք պետք է վերադարձնեք 0:
- (3) `add_teleporter(5, 3)`. Դուք պետք է վերադարձնեք 0:
- (4) `add_teleporter(1, 4)`. Այս պահին հնարավոր է ստանալ անվերջ քանակությամբ կնիքներ: Օրինակ, տուրիստը սկսում է 0 մոլորակից և անցնում է 1, 4, 5, 0, 1, 4, 5, 0, ... մոլորակներով: Այսպիսով, դուք պետք է վերադարձնեք 1, և ձեր ծրագիրը կավարտվի:

Այս օրինակը պատկերված է հետևյալ նկարում: Հատուկ մոլորակները և սկզբնական տելեպորտերները ցույց են տրված հաստ գծերով և շրջաններով: `add_teleporter`-ով ավելացվող տելեպորտերները նշված են սև քառակուսիներում սպիտակ 0-ից 4 թվերով, ըստ հերթականության:



Օրինակ 2

Դիտարկենք հետևյալ կանչը.

```
init(4, 2)
```

Այս օրինակում կա 4 մոլորակ և 2 հատուկ մոլորակ: 0 և 1 մոլորակները հատուկ մոլորակներ են: (0, 1)-ը սկզբնական տելեպորտեր է:

Ենթադրենք գրեյդերն անում է հետևյալ կանչը.

- `add_teleporter(1, 1)`. (1, 1) տելեպորտերն ավելացնելուց հետո հնարավոր է ստանալ անթիվ քանակությամբ կնիքներ: Օրինակ, տուրիստը սկսում է 1 մոլորակից և գնում է 1 մոլորակը անվերջ անգամներ օգտագործելով (1, 1) տելեպորտերը: Այսպիսով, դուք պետք է վերադարձնեք 1, և ձեր ծրագիրը կավարտվի:

Մուտքի/ելքի այլ օրինակներ կարող եք գտնել կցված փաթեթում:

Սահմանափակումներ

- $1 \leq n \leq 300\,000$
- $1 \leq m \leq 500\,000$
- $1 \leq k \leq n$

`add_teleporter` ենթածրագրի յուրաքանչյուր կանչի համար

- $0 \leq u \leq n - 1$ և $0 \leq v \leq n - 1$
- Նախքան (u, v) -ն ավելացնելը u մոլորակից v մոլորակ տանող տելեպորտեր չկա:

Ենթախնդիրներ

1. (2 միավոր) $n = k, n \leq 100, m \leq 300$
2. (10 միավոր) $n \leq 100, m \leq 300$
3. (18 միավոր) $n \leq 1\,000, m \leq 5\,000$
4. (30 միավոր) $n \leq 30\,000, m \leq 50\,000, k \leq 1\,000$
5. (40 միավոր) Լրացուցիչ սահմանափակումներ չկան:

Գրեյդերի նմուշ

Գրեյդերի նմուշը մուտքային տվյալները կարդում է հետևյալ ձևաչափով.

- տող 1. $n\ m\ k$
- տող $2 + i$ ($0 \leq i \leq m - 1$). $u[i]\ v[i]$

Գրեյդերի նմուշը սկզբում կանչում է `init` ֆունկցիան, իսկ հետո կատարում է `add_teleporter`-ի կանչեր $u = u[i]$ և $v = v[i]$ համար, որտեղ $i = 0, 1, \dots, m - 1$:

Այն տպում է 1 վերադարձնող առաջին `add_teleporter`-ի կարգահամարը (որը 0-ից $m - 1$ սահմաններում է, ներառյալ ծայրակետերը), կամ m , եթե `add_teleporter`-ի բոլոր կանչերը վերադարձնում են 0:

Եթե `add_teleporter`-ի որևէ կանչ վերադարձնում է 0-ից և 1-ից տարբեր թիվ, գրեյդերի նմուշը տպում է -1 և ձեր ծրագիրն անմիջապես ավարտվում է: