



## Миңдеген Аралдар

Миңдеген аралдар - Ява деңизинде жайгашкан кооз аралдардын тобу. Ал 0дөн  $N - 1$ ге чейин номерленген  $N$  аралдан турат.

Аралдар арасында сүзүү үчүн колдонула турган 0дөн  $M - 1$ ге чейинки  $M$  кайык бар. Ар бир  $i$  үчүн  $0 \leq i \leq M - 1$ ,  $i$ – кайык  $U[i]$  же  $V[i]$  аралына коюлуп,  $U[i]$  жана  $V[i]$  аралдардын арасында сүзүү үчүн колдонулушу мүмкүн. Тактап айтканда, кайык  $U[i]$  аралына токтогондо, аны  $U[i]$  аралынан  $V[i]$  аралына сүзүү үчүн колдонсо болот, андан кийин кайык  $V[i]$  аралына токтойт. Ошо сыяктуу эле, кайык  $V[i]$  аралына токтогондо, аны  $V[i]$  аралынан  $U[i]$  аралына сүзүү үчүн колдонсо болот, андан кийин кайык  $U[i]$  аралына токтойт.

Башында, кайык  $U[i]$  аралында болот. Бир эле жуп аралдын ортосунда сүзүү үчүн бир нече кайык колдонулушу мүмкүн. Бир эле аралга бир нече кайык конушу да мүмкүн.

Коопсуздук себептеринен улам, кайык ар бир сүзүлгөндөн кийин сакталышы керек, бул бир эле кайыкты эки жолу катары менен сүзүүгө тыюу салат. Башкача айтканда,  $i$ – кайык колдонулгандан кийин,  $i$ – кайык кайра колдонулаардан мурун дагы бир кайык колдонулушу керек.

Бу Денглек кээ бир аралдарды аралап саякаттоону каалайт. Эгерде төмөнкү шарттар аткарылса гана анын саякаты **жарактуу** деп саналат.

- Ал 0– аралдан сапарын баштап, 0– аралдан аяктайт.
- Ал 0– аралдан башка жок дегенде бир аралга барыш керек.
- Саякат аяктагандан кийин, ар бир кайык сапарга чейин кайсыл аралда болсо, ошол эле аралга келиш керек. Башкача айтканда,  $i$ – кайык , ар бир  $i$  үчүн  $0 \leq i \leq M - 1$ ,  $U[i]$  аралына орнотулушу керек.

Бу Денглекке эң көп 2 000 000 жолу сүзүү менен жарактуу жолду табууга жардам бериңиз же андай жарактуу саякат жок экенин аныктаңыз. Бул тапшырмада көрсөтүлгөн чектөөлөрдүн астында (Чектөөлөр бөлүмүн караңыз), эгерде жарактуу саякат бар болсо, анда 2 000 000дон ашык сүзүүнү талап кылбаган жарактуу саякат да бар экенин далилдесе болот.

## Ишке ашыруу маалыматтары

Сиз төмөнкү процедураны ишке ашыруңуз керек:

```
union(bool, int[]) find_journey(int N, int M, int[] U, int[] V)
```

- $N$ : аралдардын саны.
- $M$ : кайыктардын саны.
- $U, V$ : кайыктарды көрсөткөн  $M$  узундуктагы массив.
- Бул процедура `boolean` же бүтүн сандардын массивин жибереш керек.
  - Эгерде жарактуу саякат жок болсо, процедура `"false"` жибереш керек.
  - Эгер жарактуу саякат болсо, сизде эки вариант бар:
    - Толук упайга ээ болуу үчүн процедура жарактуу жолду билдирген эң көп 2 000 000 бүтүн сан массивин жибереш керек. Тагыраак айтканда, бул массивдин элементтери саякатта колдонулган кайыктардын номерлери болушу керек (алар колдонулган тартипте).
    - Жарым-жартылай упайды алуу үчүн процедура `"true"`, 2 000 000 ашык бүтүн сандардын массивин же жарактуу жолду сүрөттөбөгөн бүтүн сандардын массивин жибереш керек. (Көбүрөөк маалымат үчүн Кошумча тапшырмалар бөлүмүн караңыз.)
- Бул процедура так бир жолу чакырылат.

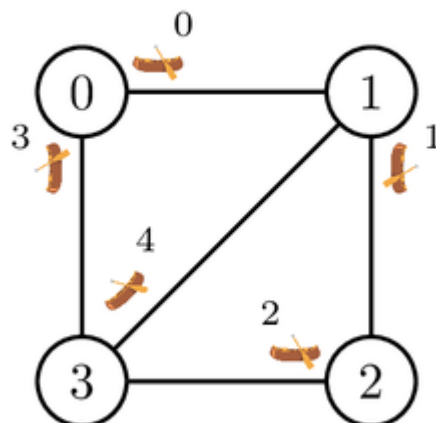
## Мисалдар

### 1-мисал

Төмөнкү чакырууну карап көрөлү:

```
find_journey(4, 5, [0, 1, 2, 0, 3], [1, 2, 3, 3, 1])
```

Аралдар жана кайыктар төмөндөгү сүрөттө көрсөтүлгөн.



Мүмкүн болгон жарактуу саякат төмөнкүдөй. Бу Денглек биринчи иретте кайык менен 0, 1, 2 жана 4 сүзөт. Натыйжада, ал 1 аралында турат. Андан кийин Бу Денглек кайык менен 0 сүзө

алат, анткени ал учурда 1 аралында жайгашкан жана ал колдонгон акыркы кайык 0 эмес. Кайра кайык 0 сүзгөндөн кийин, Бу Денглек азыр 0 аралында. Бирок 1, 2 жана 4 каноэлери саякатка чейинки аралдарга коюлбайт. Андан кийин Бу Денглек 3, 2, 1, 4 жана дагы 3 кайык сүзүү менен сапарын улантат. Бу Денглек 0 аралына кайтып келди жана бардык кайыктар саякатка чейинки аралдарга орнотулган.

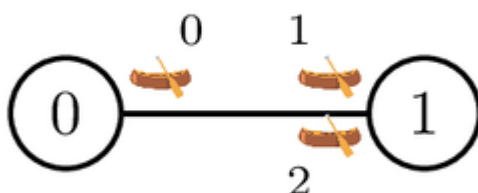
Демек, кайтарылган маани  $[0, 1, 2, 4, 0, 3, 2, 1, 4, 3]$  жарактуу саякатты билдирет.

## 2-мисал

Төмөнкү чакырууну карап көрөлү:

```
find_journey(2, 3, [0, 1, 1], [1, 0, 0])
```

Аралдар жана кайыктар төмөндөгү сүрөттө көрсөтүлгөн.



Бу Денглек каноэ менен 0 гана баштай алат, андан кийин ал 1– же 2– кайыкта сүзө алат. Белгилей кетсек, ал 0–кайыкты катары менен эки жолу сүзө албайт. Эки учурда тең, Бу Денглек аралга 0– аралга кайтып келди. Бирок, кайыктар саякатка чейинки аралдарга коюлбайт жана Бу Денглек андан кийин эч кандай кайыкта сүзө албайт, анткени 0 аралына конгон жалгыз кайык ал жаңы эле колдонгон кайык. Жарактуу жол жок болгондуктан, процедура "false" деп кайтарылышы керек.

## Чектөөлөр

- $2 \leq N \leq 100\,000$
- $1 \leq M \leq 200\,000$
- $0 \leq U[i] \leq N - 1$  жана  $0 \leq V[i] \leq N - 1$  (ар бир  $i$  үчүн  $0 \leq i \leq M - 1$ )
- $U[i] \neq V[i]$  (ар бир  $i$  үчүн  $0 \leq i \leq M - 1$ )

## Кошумча тапшырмалар

1. (5 упай)  $N = 2$

2. (5 упай)  $N \leq 400$ .  $x$  жана  $y$  ( $0 \leq x < y \leq N - 1$ ) ар бир жуп аралы үчүн алардын ортосунда сүзүү үчүн колдонула турган так эки кайык бар. Алардын бири  $x$  аралына, экинчиси  $y$  аралына орнотулган.
3. (21 упай)  $N \leq 1000$ ,  $M$  жуп жана ар бир **жуп**  $i$  үчүн  $0 \leq i \leq M - 1$ , кайык  $i$  жана  $i + 1$  экөөнү тең  $U[i]$  жана  $V[i]$  аралдарынын ортосунда сүзүү үчүн колдонсо болот.  $i$  кайык  $U[i]$  аралына, ал эми кайык  $i + 1$   $V[i]$  аралына орнотулган. Формалдуу түрдө  $U[i] = V[i + 1]$  жана  $V[i] = U[i + 1]$ .
4. (24 упай)  $N \leq 1000$ ,  $M$  жуп жана ар бир **жуп**  $i$  үчүн  $0 \leq i \leq M - 1$ , кайык  $i$  жана  $i + 1$  экөөнү тең  $U[i]$  жана  $V[i]$  аралдарынын ортосунда сүзүү үчүн колдонсо болот. Эки кайык тең  $U[i]$  аралына орнотулган. Формалдуу түрдө  $U[i] = U[i + 1]$  жана  $V[i] = V[i + 1]$ .
5. (45 балл) Кошумча чектөөлөр жок.

Жарактуу саякат бар болгон ар бир сыноо үчүн сиздин чечимиңиз:

- жарактуу жолду кайтарса, толук упай алат,
- true, 2 000 000 ашык бүтүн сандардын массивин же жарактуу саякатты сүрөттөбөгөн массивди кайтарса, 35% упай алат,
- антпесе 0 упай алат.

Жарактуу саякат жок болгон ар бир сыноо үчүн сиздин чечимиңиз:

- false деп кайтарса, толук упай алат,
- антпесе 0 упай алат.

Ар бир көмөкчү тапшырма үчүн жыйынтык упай чакан тапшырмадагы тесттик иштердин минималдуу упайлары экенин белгилеңиз.

## Үлгү грейдер

Үлгү грейдер киргизүүнү төмөнкү форматта окуйт:

- 1– сап :  $N M$
- Кийинки  $2 + i$  сапка ( $0 \leq i \leq M - 1$ ):  $U[i] V[i]$

Үлгү грейдер жоопторуңузду төмөнкү форматта басып чыгарат:

- Эгерде 'find\_journey' 'bool' кайтарса:
  - 1– сап: 0
  - 2– сапка: 0, эгерде find\_journey false кайтарса, же болбосо 1.
- Эгерде find\_journey int[] кайтарса, бул массивдин элементтерин  $c[0], c[1], \dots, c[k - 1]$  менен белгилеңиз. Үлгү грейдер басып чыгарат:
  - 1– сап: 1
  - 2– сап :  $k$
  - 3– сап:  $c[0] c[1] \dots c[k - 1]$