

## Շատրվանների այգի

Մոտակա այգում կան  $n$  շատրվաններ, համարակալված  $0$ -ից  $n - 1$  թվերով: Մենք մոդելավորում ենք շատրվանները, որպես կետեր երկչափ հարթության վրա: Այսինքն,  $i$ -րդ շատրվանը ( $0 \leq i \leq n - 1$ ) ( $x[i], y[i]$ ) կետն է, որտեղ  $x[i]$ -ն և  $y[i]$ -ն **զույգ ամբողջ** թվեր են: Բոլոր շատրվանների դիրքերը իրարից տարբեր են:

Ճարտարապետ Թիմըթին վարձվել է ինչ-որ **ճանապարհներ** կառուցելու և մեկ **ստտարան** յուրաքանչյուր ճանապարհի վրա տեղադրելու համար: Ճանապարհը  $2$  երկարությամբ **հորիզոնական** կամ **ուղղահայաց** գծի հատված է, որի ծայրակետերը երկու տարբեր շատրվաններ են: Ճանապարհները պետք է կառուցվեն այնպես, որ հնարավոր լինի ճանապարհորդել կամայական երկու շատրվանների միջև շարժվելով ճանապարհների երկայնքով: Ի սկզբանե այգում ճանապարհներ չկան:

Յուրաքանչյուր ճանապարհի համար, **ճիշտ** մեկ ստտարան անհրաժեշտ է տեղադրվի այգում և **վերագրվի** այդ ճանապարհին (այսինքն երեսը դեպի ճանապարհը): Յուրաքանչյուր ստտարան պետք է տեղադրվի ինչ-որ  $(a, b)$  կետում այնպես, որ  $a$ -ն և  $b$ -ն **կենտ ամբողջ** թվեր են: Բոլոր ստտարանների դիրքերը պետք է լինեն իրարից տարբեր:  $(a, b)$ -ի ստտարանը կարող է վերագրվել միայն այն ճանապարհին, եթե այդ ճանապարհի **երկու** ծայրակետերն էլ  $(a - 1, b - 1)$ ,  $(a - 1, b + 1)$ ,  $(a + 1, b - 1)$  և  $(a + 1, b + 1)$  թվում են: Օրինակ,  $(3, 3)$ -ի ստտարանը կարող է վերագրվել միայն այն ճանապարհին, որը այս չորս գծի հատվածներից մեկն է՝  $(2, 2) - (2, 4)$ ,  $(2, 4) - (4, 4)$ ,  $(4, 4) - (4, 2)$ ,  $(4, 2) - (2, 2)$ .

Օգնեք Թիմըթին որոշելու համար հնարավոր է արդյոք կառուցել ճանապարհներ, և տեղադրել և վերագրել ստտարաններ այնպես, որ բավարարվեն բոլոր վերը նշված պայմանները, և եթե այո, ապահովել նրան հնարավոր լուծում: Եթե կան մեկից ավելի հնարավոր լուծումներ, որոնք բավարարում են բոլոր պայմաններին, դուք կարող եք հաղորդել նրանցից ինչ-որ մեկը:

## Իրականացման մանրամասներ

Դուք պետք է իրականացնեք հետևյալ ֆունկցիան.

```
int construct_roads(int[] x, int[] y)
```

- $x, y$ : երկու  $n$  երկարությամբ զանգվածներ: Յուրաքանչյուր  $i$ -ի համար ( $0 \leq i \leq n - 1$ ),  $i$ -րդ շատրվանը ( $x[i], y[i]$ ) կետն է, որտեղ  $x[i]$ -ն և  $y[i]$  are even integers.
- Եթե կառուցումը հնարավոր է, ֆունկցիան պետք է կատարի `build`-ի ճիշտ մեկ կանչ (տեսքը ներքևում) հաղորդելու համար լուծումը, որից հետո պետք է

վերադարձնի 1:

- Հակառակ դեպքում, ֆունկցիան պետք է վերադարձնի 0 առանց build-ի ինչ-որ կանչ կատարելու:
- Այս ֆունկցիան կանչվում է ճիշտ մեկ անգամ:

Ձեր իրականացումը կարող է կանչել հետևյալ ֆունկցիան ապահովելով հնարավոր ճանապարհների կառուցում և նստարանների տեղադրում.

```
void build(int[] u, int[] v, int[] a, int[] b)
```

- $m$ -ը կառուցման մեջ ընդհանուր ճանապարհների քանակն է:
- $u, v$ : երկու  $m$  երկարությամբ զանգվածներ, ներկայացնող կառուցված ճանապարհները: Այդ ճանապարհները համարակալված են 0-ից  $m - 1$ -ով: Յուրաքանչյուր  $j$ -ի համար ( $0 \leq j \leq m - 1$ ),  $j$ -րդ ճանապարհը միացնում է  $u[j]$  և  $v[j]$  շատրվանները: Յուրաքանչյուր ճանապարհ պետք է լինի 2 երկարությամբ հորիզոնական կամ ուղղահայաց գծի հատված: Կամայական երկու տարբեր ճանապարհներ կարող են ունենալ ամենաշատը մեկ ընդհանուր կետ (շատրվան): Երբ ճանապարհները կառուցվեն, պետք է հնարավոր լինի ճանապարհորդել կամայական երկու շատրվանների միջև շարժվելով ճանապարհների երկայնքով:
- $a, b$ : երկու  $m$  երկարությամբ զանգվածներ, ներկայացնող նստարանները: Յուրաքանչյուր  $j$ -ի համար ( $0 \leq j \leq m - 1$ ), նստարանը տեղադրված է  $(a[j], b[j])$ -ում, և վերագրված է  $j$ -րդ ճանապարհին: Ոչ մի երկու տարբեր նստարաններ չպետք է ունենան նույն դիրքը:

## Օրինակներ

### Օրինակ 1

Ենթադրենք հետևյալ կանչը.

```
construct_roads([4, 4, 6, 4, 2], [4, 6, 4, 2, 4])
```

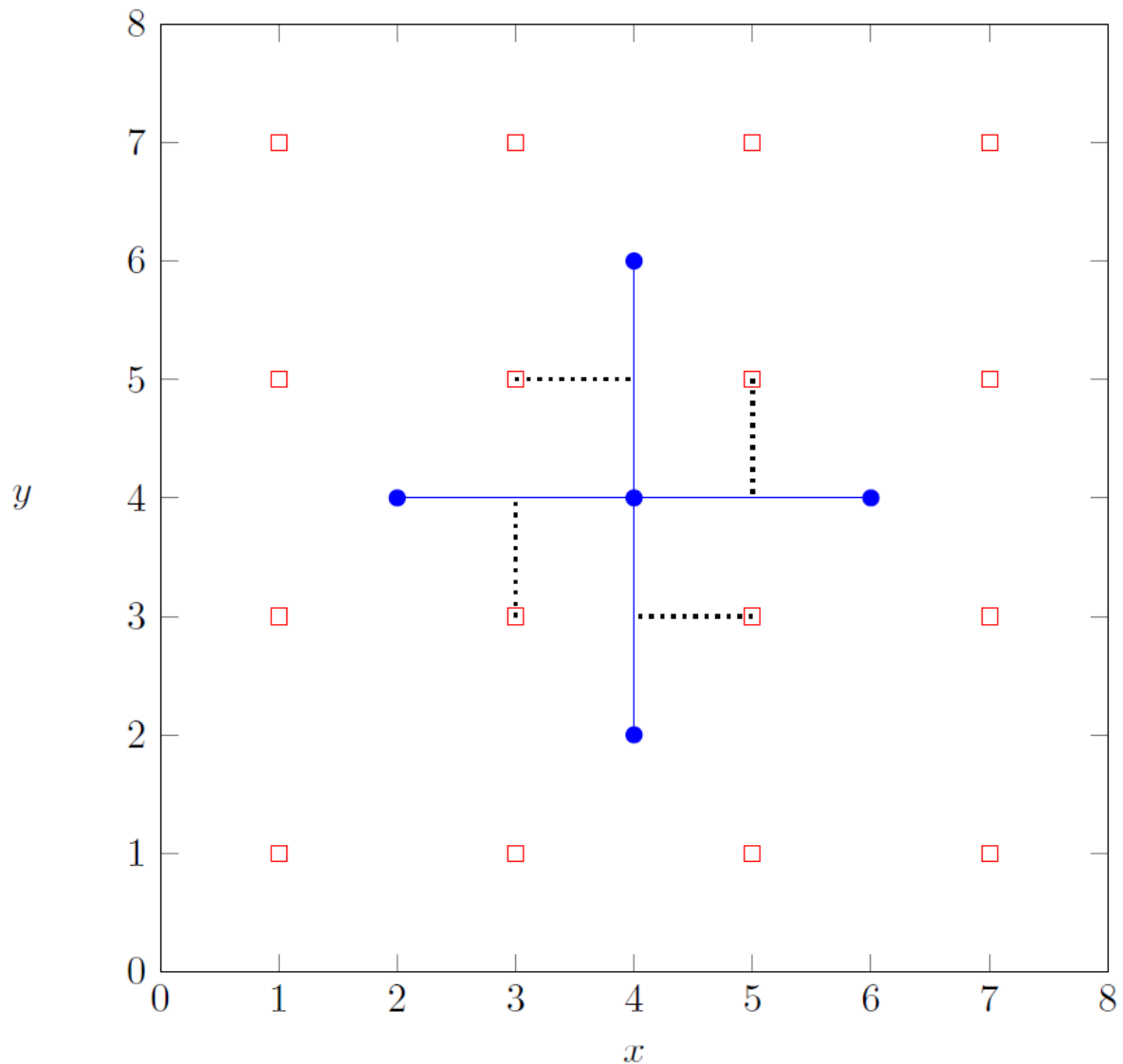
Սա նշանակում է, որ կան 5 շատրվաններ.

- 0-րդ շատրվանը տեղադրված է (4, 4)-ում,
- 1-ին շատրվանը տեղադրված է (4, 6)-ում,
- 2-րդ շատրվանը տեղադրված է (6, 4)-ում,
- 3-րդ շատրվանը տեղադրված է (4, 2)-ում,
- 4-րդ շատրվանը տեղադրված է (2, 4)-ում:

Հնարավոր է կառուցել հետևյալ 4 ճանապարհները, որտեղ յուրաքանչյուր ճանապարհ միացնում է երկու շատրվաններ, և տեղադրել համապատասխան նստարաններ.

Ճանապարհի համար	Ճանապարհի միացնող շատրվանի համարները	Վերագրված նստարանի դիրքը
0	0, 2	(5, 5)
1	0, 1	(3, 5)
2	3, 0	(5, 3)
3	4, 0	(3, 3)

Այս լուծումը համապատասխանում է հետևյալ դիագրամին.



Հաղորդելու համար այս լուծումը, `construct_roads` պետք է կատարի հետևյալ կանչը.

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

Հետո պետք է վերադարձնել 1:

Նկատեք, որ այս դեպքում, կան մեկից ավելի լուծումներ, որոնք բավարարում են պահանջներին, նրանցից բոլորը կհամարվեն ճիշտ: Օրինակ, ճիշտ է նաև կանչել `build([1, 2, 3, 4], [0, 0, 0, 0], [5, 5, 3, 3], [5, 3, 3, 5])` և հետո վերադարձնել 1:

## Օրինակ 2

Ենթադրենք հետևյալ կանչը.

```
construct_roads([2, 4], [2, 6])
```

0-րդ շատրվանը տեղադրված է (2, 2)-ում, իսկ 1-ին շատրվանը տեղադրված է (4, 6)-ում: Քանի որ, չկա ինչ-որ հնարավոր տարբերակ կառուցելու համար ճանապարհներ, որոնք բավարարում են պահանջներին, `construct_roads`-ը պետք է վերադարձնի 0 առանց `build`-ի ինչ-որ կանչ կատարելու:

## Սահմանափակումներ

- $1 \leq n \leq 200\,000$
- $2 \leq x[i], y[i] \leq 200\,000$  (բոլոր  $0 \leq i \leq n - 1$  համար)
- $x[i]$  and  $y[i]$  are even integers (բոլոր  $0 \leq i \leq n - 1$  համար):
- Ոչ մի երկու շատրվան չունեն նույն դիրքը:

## Ենթախնդիրներ

1. (5 միավոր)  $x[i] = 2$  (բոլոր  $0 \leq i \leq n - 1$  համար)
2. (10 միավոր)  $2 \leq x[i] \leq 4$  (բոլոր  $0 \leq i \leq n - 1$  համար)
3. (15 միավոր)  $2 \leq x[i] \leq 6$  (բոլոր  $0 \leq i \leq n - 1$  համար)
4. (20 միավոր) Կա ամենաշատը մեկ եղանակ կառուցելու համար ճանապարհները, այնպես որ հնարավոր լինի ճանապարհորդել կամայական երկուսի միջև շարժվելով միայն ճանապարհների երկայնքով:
5. (20 միավոր) Գոյություն չունեն չորս շատրվաններ, որոնք կազմվում են  $2 \times 2$  քառակուսու անկյունները:
6. (30 միավոր) Հավելյալ սահմանափակումներ չկան:

## Գրեյդերի նմուշ

Գրեյդերի նմուշը կարդում է մուտքը հետևյալ ֆորմատով.

- line 1 :  $n$
- line  $2 + i$  ( $0 \leq i \leq n - 1$ ):  $x[i]$   $y[i]$

Գրեյդերի նմուշի ելքը հետևյալ ֆորմատով է.

- line 1: վերադարձնում է `construct_roads`-ի արժեքը

Եթե `construct_roads`-ի վերադարձվող արժեքը 1 է և `build(u, v, a, b)`-ը կանչվում է, ապա գրեյդերը հավելյալ տպում է.

- line 2:  $m$
- line  $3 + j$  ( $0 \leq j \leq m - 1$ ):  $u[j] \ v[j] \ a[j] \ b[j]$