

Паркот со фонтани

Во паркот во Пирава одлучено е да се постават n фонтани, означени со броеви од 0 до $n - 1$. Фонтаните може да се разгледуваат како точки на (дво-димензионална) рамнина. Имено, фонтаната i ($0 \leq i \leq n - 1$) е точка $(x[i], y[i])$ каде $x[i]$ и $y[i]$ се **парни цели броеви**. Сите фонтани се поставени на различни позиции.

Архитектот Јованов е одговорен да го испланира конструирањето на некои **патеки** и поставувањето на по една **клуба** на секоја од патеките. Патека претставува **хоризонтална** или **вертикална** отсечка со должина 2, чии што крајни точки се две различни фонтани. Патеките треба да бидат конструирани така да може да се стигне од било која фонтана до било која друга фонтана движејќи се по патеките. Иницијално, нема ниту една патека во паркот.

За секоја патека, **точно** една клуба треба да биде поставена во паркот и да биде **доделена** на таа патека (т.е. клубата да биде завртена кон патеката). Секоја клуба мора да биде поставена на некоја точка (a, b) така да a и b се **непарни цели броеви**. Сите клуби мора да бидат поставени на различни позиции. Клуба поставена на позиција (a, b) може да биде доделена на патека ако **и двете** од крајните точки на патеката се некои од $(a - 1, b - 1)$, $(a - 1, b + 1)$, $(a + 1, b - 1)$ и $(a + 1, b + 1)$. На пример, клубата на позиција $(3, 3)$ може да биде доделена на патека, ако патеката е една од следните четири отсечки $(2, 2) - (2, 4)$, $(2, 4) - (4, 4)$, $(4, 4) - (4, 2)$, $(4, 2) - (2, 2)$.

Помогнете му на Јованов да одлучи дали е можно да се конструираат патеки, и да се постават и доделат клуби задоволувајќи ги сите услови дадени погоре, и доколку е тоа можно најдете едно такво решение. Доколку има повеќе решенија кои ги задоволуваат условите дадете било кое едно од нив.

Детали за имплементација

Вие треба да ја имплементирате следната процедура.

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

- x, y : две низи со должина n . За секое i ($0 \leq i \leq n - 1$), фонтаната i е точка $(x[i], y[i])$, каде $x[i]$ и $y[i]$ се позитивни цели броеви.
- Доколку постои решение, оваа процедура треба да направи точно еден повик на `build` процедурата (видете подолу) за да го предадете решението, после што треба да врати 1.
- Инаку, процедурата треба да врати 0 без да прави повици на `build`.
- Оваа процедура е повикана точно еднаш.

Вашата имплементација може да ја повика следната процедура за да предадете решение, односно конструкција на патеките и поставување на клупи на нив т.д. се исполнети бараните услови:

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

- Нека m е тоталниот број на патеки кои ќе ги конструирате.
- u, v : две низи со должина m , кои ги претставуваат патеките кои ќе бидат конструирани. Овие патеки се означени од 0 до $m - 1$. За секој j ($0 \leq j \leq m - 1$), патеката j ги поврзува фонтаните $u[j]$ и $v[j]$. Една патека мора да е хоризонтална или вертикална отсечка со должина 2 . Било кои две различни патеки може да имаат најмногу една заедничка точка (фонтана). Кога ќе бидат конструирани сите патеки, мора да е можно да се патува помеѓу било кои две фонтани движејќи се по конструираниите патеки.
- a, b : две низи со должини m , кои ги претставуваат клупите. За секој j ($0 \leq j \leq m - 1$), поставена е клупа на $(a[j], b[j])$, и е доделена на патеката j . Две различни клупи не може да бидат поставени на иста позиција.

Примери

Пример 1

Да го разгледаме следниот повик:

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

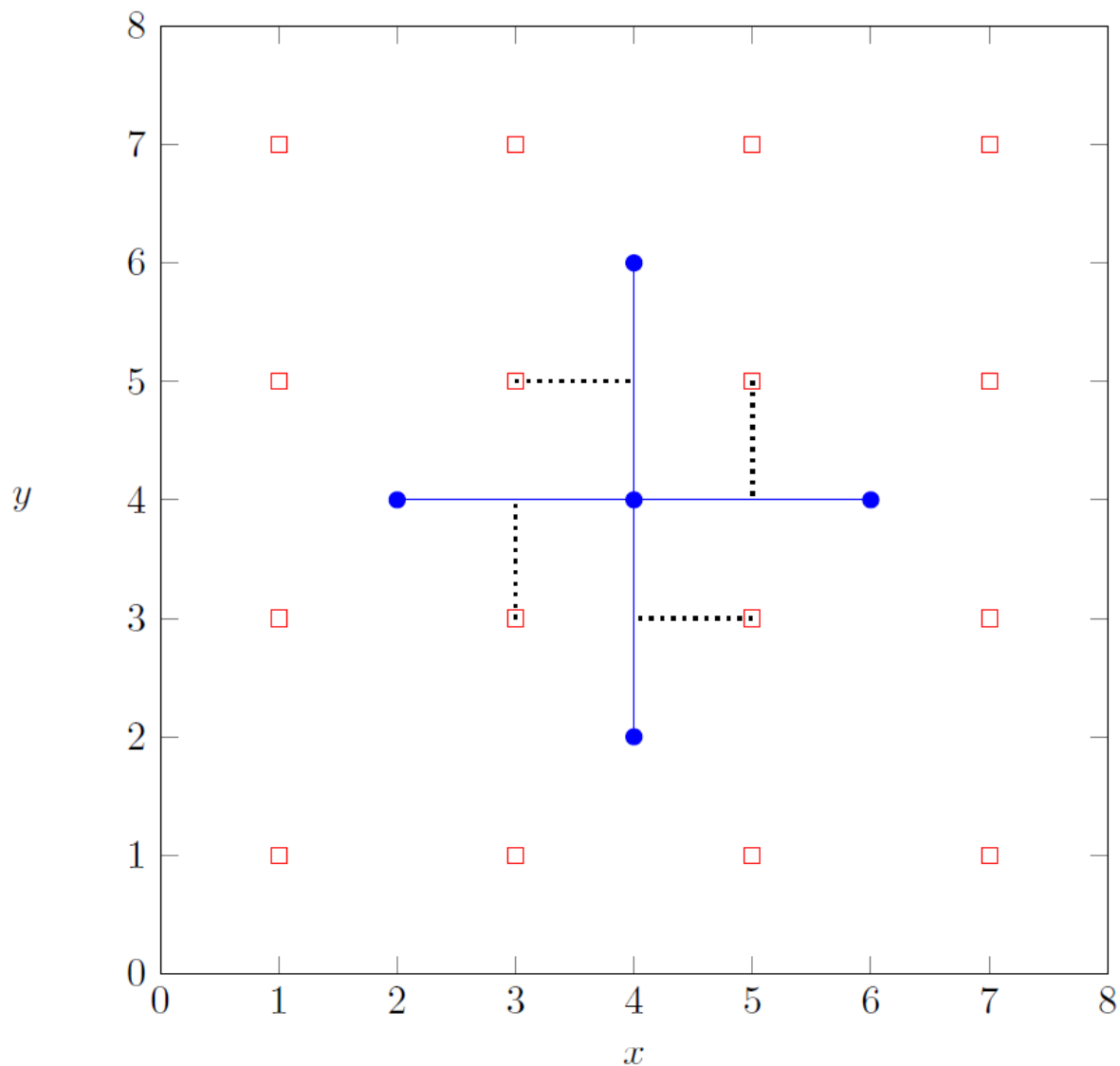
Ова значи дека има 5 фонтани:

- фонтаната 0 е поставена на позиција $(4, 4)$,
- фонтаната 1 е поставена на позиција $(4, 6)$,
- фонтаната 2 е поставена на позиција $(6, 4)$,
- фонтаната 3 е поставена на позиција $(4, 2)$,
- фонтаната 4 е поставена на позиција $(2, 4)$.

Можно е да се конструираат следните 4 патеки, каде секоја патека поврзува две фонтани, и да се постават соодветните клупи:

Ознака на патеката	Ознака на фонтаните кои патеката ги поврзува	Позиција на доделените клупи
0	0, 2	(5, 5)
1	0, 1	(3, 5)
2	3, 0	(5, 3)
3	4, 0	(3, 3)

Решението соодветува на следниот дијаграм:



За да го предадете ова решение, `construct_roads` треба да го направи следниот повик:

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

Потоа треба да терминаира враќајќи 1.

Забележете дека во овој случај, постојат повеќе решенија кои ги задоволуваат сите услови, било кое едно од нив би се сметало за точно. На пример, исто така е точно да се повика `build([1, 2, 3, 4], [0, 0, 0, 0], [5, 5, 3, 3], [5, 3, 3, 5])` и потоа да се врати 1.

Пример 2

Да го разгледаме следниот повик:

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

Фонтаната 0 е поставена на позиција $(2, 2)$ и фонтаната 1 е поставена на позиција $(4, 6)$. Бидејќи не постои начин да се конструираат патеки кои ги задоволуваат условите, `construct_roads` треба да врати 0 без да прави повик на `build`.

Ограничувања

- $1 \leq n \leq 200\,000$
- $2 \leq x[i], y[i] \leq 200\,000$ (за сите $0 \leq i \leq n - 1$)
- $x[i]$ и $y[i]$ се позитивни цели броеви (за сите $0 \leq i \leq n - 1$).
- Нема две фонтани со иста позиција

Подзадачи

1. (5 поени) $x[i] = 2$ (за сите $0 \leq i \leq n - 1$)
2. (10 поени) $2 \leq x[i] \leq 4$ (за сите $0 \leq i \leq n - 1$)
3. (15 поени) $2 \leq x[i] \leq 6$ (за сите $0 \leq i \leq n - 1$)
4. (20 поени) Има најмногу еден начин да се конструираат патеките, така да може да се патува меѓу било кои две фонтани движејќи се по патеки.
5. (20 поени) Не постојат четири фонтани кои ги формираат темињата на 2×2 квадрат.
6. (30 поени) Нема дополнителни ограничувања.

Оценувач

Дадениот оценувач го чита влезот во следниот формат:

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

Излезот од дадениот оценувач е во следниот формат:

- ред 1: вредноста вратена од `construct_roads`

Ако вратената вредност од `construct_roads` е 1 и `build(u, v, a, b)` е повикана, оценувачот дополнително ќе испечати:

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