

Fountain Parks

Veliki Milan postavio je n **fontana** usred Zagreba te ih označio brojevima od 0 do $n - 1$. Fontane je još za života modelirao točkama u dvodimenzionalnoj ravnini. Preciznije, i -tu ($0 \leq i \leq n - 1$) je fontanu predstavio točkom $(x[i], y[i])$, gdje su $x[i]$ i $y[i]$ parni brojevi ili 0 . Dakako, lokacije svih fontana međusobno su različite.

Milanu u čast, Tomislav je odlučio izgraditi neke **puteljke** i na svakom puteljku odlučio je postaviti jednu **klupicu** gdje će moderni zagrepčani moći odmoriti nakon naporne vožnje biciklom ili šetnje sa psom. Svaki puteljak je **horizontalna** ili **vertikala** dužina duljine 2 , čije krajnje točke odgovaraju dvjema različitim fontanama. Puteljci trebaju biti izgrađeni tako da se može proputovati između bilo koje dvije fontane koristeći jedan ili više novoizgrađenih puteljaka. Početno stanje je takvo da nije izgrađen niti jedan puteljak.

Na svakom puteljku treba biti postavljena **točno jedna** klupica. Svaka klupica treba biti postavljena na neku točku (a, b) takvu da su a i b neparni brojevi. Lokacije klupica moraju međusobno biti **različite**. Klupica na poziciji (a, b) može biti dodijeljena nekom puteljku ako su **obje** krajnje točke puteljka među točkama $(a - 1, b - 1)$, $(a - 1, b + 1)$, $(a + 1, b - 1)$ and $(a + 1, b + 1)$. Primjerice, kupica u točki $(3, 3)$ može biti dodijeljena samo puteljcima koji su predstavljeni dužinama $(2, 2) - (2, 4)$, $(2, 4) - (4, 4)$, $(4, 4) - (4, 2)$, $(4, 2) - (2, 2)$.

Iako Tomislav uporno ponavlja "ZNAM DA MOŽEMO", došlo je vrijeme da priupita nekog tko stvarno zna odgovoriti na pitanje možemo li izgraditi puteljke i klupice u gradu Zagrebu na način kako je gore opisano. Ako možemo, Tomislava zanima i kako bismo to mogli napraviti. Ako postoji više valjanih načina za izgradnju puteljaka i klupica, dovoljno mu je dojaviti bilo koje.

Implementacijski detalji

Potrebno je implementirati sljedeću proceduru:

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

- x, y : dva polja duljine n . Za svaki i ($0 \leq i \leq n - 1$), fontana i predstavljena je točkom $(x[i], y[i])$, gdje su $x[i]$ i $y[i]$ parni brojevi ili 0 .
- Ako je konstrukcija moguća, ova procedura treba točno jednom pozvati proceduru `build` (detaljnije objašnjeno u nastavku teksta) kako bi dojavila rješenje. Nakon tog poziva, procedura treba vratiti `1`.
- Inače, procedura treba vratiti `0` bez poziva procedure `build`.
- Ova će procedura biti pozvana točno jednom.

Vaša implementacija može (treba) pozvati sljedeću proceduru kako bi dojavila ispravnu konstrukciju puteljaka i postave klupica:

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

- u, v : dva polja duljine m , koja označavaju puteljke. Puteljci su označeni brojevima od 0 do $m - 1$. Za svaki j ($0 \leq j \leq m - 1$), puteljak j povezuje fontane $u[j]$ i $v[j]$. Svaki puteljak mora biti horizontalna ili vertikalna dužina duljine 2. Svaka dva različita puteljka smiju imati najviše jednu zajedničku točku (fontanu). Nakon što su puteljci izgrađeni, treba biti moguće proputovati između svake dvije fontane koristeći jedan ili više spomenutih puteljaka.
- a, b : dva polja duljine m , koja označavaju klupice. Za svaki j ($0 \leq j \leq m - 1$), klupica je postavljena u točki $(a[j], b[j])$ i pripada puteljku j . Niti jedne dvije klupice ne smiju biti na istoj lokaciji, a različite klupice ne smiju pripadati istom puteljku.

Primjeri

Primjer 1

Promotrimo sljedeći poziv:

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

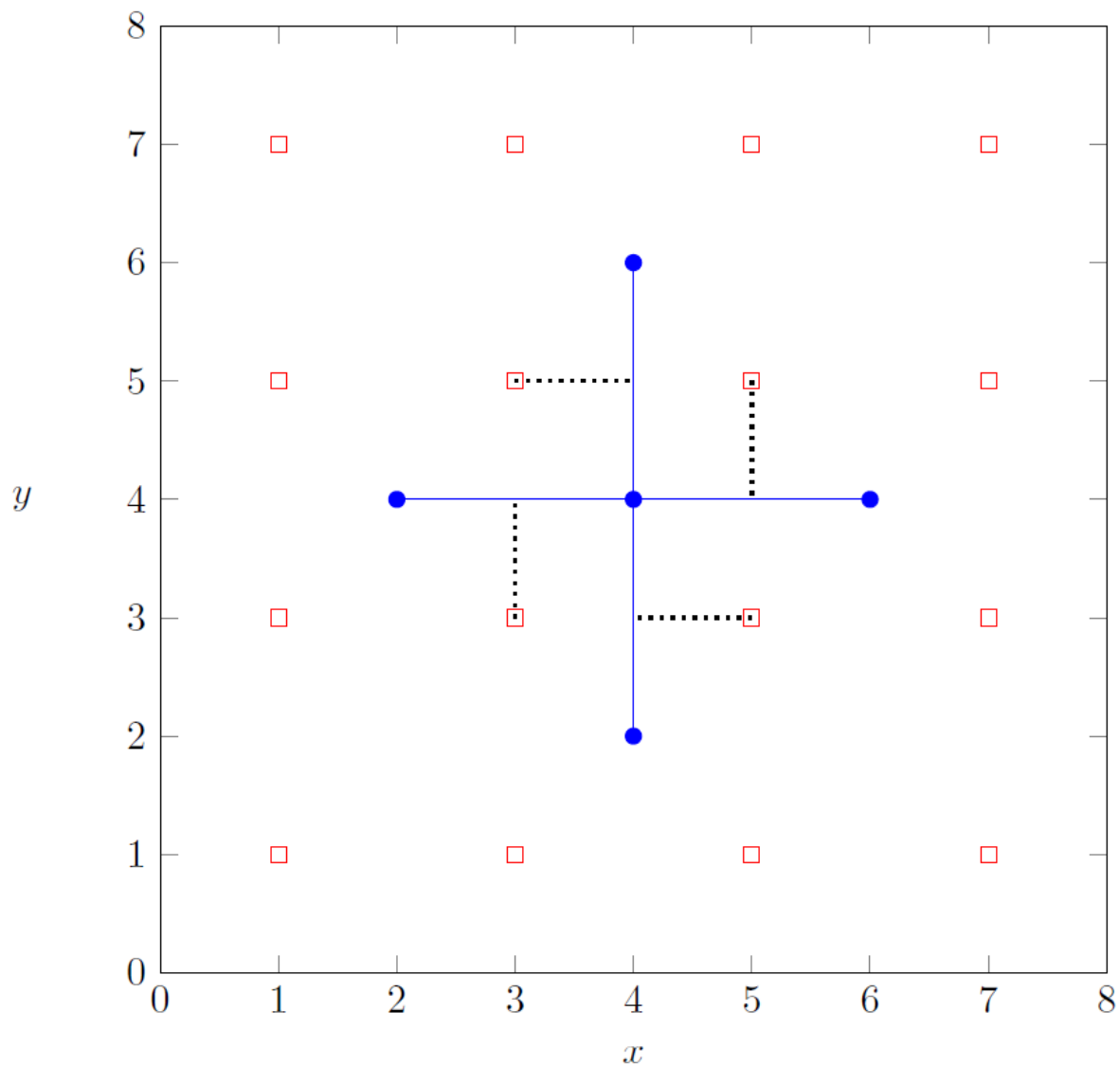
Odnosno, postoji 5 fontana:

- fontana 0 se nalazi u točki (4, 4),
- fontana 1 se nalazi u točki (4, 6),
- fontana 2 se nalazi u točki (6, 4),
- fontana 3 se nalazi u točki (4, 2),
- fontana 4 se nalazi u točki (2, 4).

Moguće je izgraditi sljedeća 4 puteljka, gdje svaki puteljak spaja dvije fontane i svakom puteljku pripadaju sljedeće klupice:

Oznaka puteljka	Oznake fontana koje spaja puteljak	Lokacija pripadajuće klupice
0	0, 2	(5, 5)
1	0, 1	(3, 5)
2	3, 0	(5, 3)
3	4, 0	(3, 3)

Ovo rješenje odgovara sljedećem dijagramu:



Da biste dojavili to rješenje, `construct_roads` treba obaviti sljedeći poziv:

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

Nakon toga treba vratiti 1.

Primijetite da u ovom slučaju postoji više rješenja koja zadovoljavaju sve uvjete. Svako takvo rješenje smatra se ispravnim. Primerice, ispravno je pozvati `build([1, 2, 3, 4], [0, 0, 0, 0], [5, 5, 3, 3], [5, 3, 3, 5])` i tada vratiti 1.

Primjer 2

Promotrimo sljedeći poziv:

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

Fontana 0 nalazi se u točki (2,2), a fontana 1 se nalazi u točki (4,6). Budući da ne postoji način za izgradnju puteljaka koji zadovoljava uvjete iz teksta zadatka, `construct_roads` treba vratiti 0

bez pozivanja procedure `build`.

Ograničenja

- $1 \leq n \leq 200\,000$
- $2 \leq x[i], y[i] \leq 200\,000$ (za svaki $0 \leq i \leq n - 1$)
- $x[i]$ i $y[i]$ su parni brojevi ili 0 (za svaki $0 \leq i \leq n - 1$).
- Niti jedne dvije fontane se ne nalaze na istoj lokaciji.

Podzadaci

1. (5 bodova) $x[i] = 2$ (za svaki $0 \leq i \leq n - 1$)
2. (10 bodova) $2 \leq x[i] \leq 4$ (za svaki $0 \leq i \leq n - 1$)
3. (15 bodova) $2 \leq x[i] \leq 6$ (za svaki $0 \leq i \leq n - 1$)
4. (20 bodova) Postoji najviše jedan način za izgradnju puteljaka takav da je moguće putovati između bilo koje dvije fontane.
5. (20 bodova) Ne postoje četiri fontane koje čine vrhove kvadrata dimenzija 2×2 .
6. (30 bodova) Nema dodatnih ograničenja.

Ogledni ocjenjivač

Ogledni ocjenjivač čita ulaz u sljedećem obliku:

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

Izlaz oglednog ocjenjivača je sljedećeg oblika:

- redak 1: povratna vrijednost poziva `construct_roads`

Ako je povratna vrijednost `construct_roads` jednaka 1 i `build(u, v, a, b)` je pozvan, ocjenjivač dodatno ispisuje:

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