

Усан оргилуурын цэцэрлэгт хүрээлэн

Ойролцоо паркт 0 -ээс $n - 1$ хүртэл дугаарлагдсан n **усан оргилуур** бий. Бид усан оргилууруудыг хоёр хэмжээст хавтгай дээрх цэг болгон загварчилж болно. Тухайлбал, i ($0 \leq i \leq n - 1$) дүгээр усан оргилуур $(x[i], y[i])$ цэг болох ба энд $x[i]$ ба $y[i]$ нь **тэгш тоонууд** байна. Усан оргилууруудын байршил бүгд ялгаатай.

Архитектор Тимоти **замууд** тавьж, зам бүрд нэг **вандан** сандал байрлуулахаар төлөвлөсөн. Зам нь хоёр төгсгөл нь ялгаатай усан оргилуур болох 2 урттай **хэвтээ** эсвэл **босоо** шулууны сегмент юм. Авто замууд нь дурын хоёр усан оргилуурын хооронд зам даган явж болохуйцаар хийгдсэн байх ёстой. Эхний ээлжинд цэцэрлэгт хүрээлэнгийн зам байхгүй. Замууд нь дурын хоёр усан оргилуурын хооронд замаар яваад очиж болохуйцаар хийгдсэн байх ёстой. Эхний ээлжинд цэцэрлэгт хүрээлэнд зам байхгүй байна.

Паркийн зам бүрд **яг** нэг вандан сандал байрлуулж, тухайн замд **хуваарилах** ёстой. Вандан сандал бүрийг (a, b) цэг дээр байрлана. a ба b нь **сондгой тоонууд** байна. Бүх вандан садлын байрлал **ялгаатай** байх ёстой. Хэрэв замуудын **хоёр** төгсгөл зөвхөн $(a - 1, b - 1)$, $(a - 1, b + 1)$, $(a + 1, b - 1)$ ба $(a + 1, b + 1)$ байх тохиолдолд л (a, b) байрлалд вандан сандлыг байгуулж болно. Жишээлбэл, $(3, 3)$ байрлалд вандан сандлыг $(2, 2) - (2, 4)$, $(2, 4) - (4, 4)$, $(4, 4) - (4, 2)$, $(4, 2) - (2, 2)$ гэсэн дөрвөн шулууны сегментийн аль нэг байх тэр замд байгуулж болно.

Тимотид тусалж, зам барих боломжтой эсэх болон дээр дурдсан бүх нөхцлийг хангасан вандан сандал байрлуулах, хуваарилах боломжтой эсэхийг тодорхойлж, боломжтой бол түүнд боломжит шийдийг гаргаж өгнө үү. Хэрэв бүх нөхцлийг хангасан олон боломжит шийд байгаа бол тэдгээрийн аль нэгийг гаргаж болно.

Хэрэгжүүлэлтийн мэдээлэл

Та дараах функцийг хэрэгжүүлэх ёстой:

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

- x, y : n урттай хоёр массив. i ($0 \leq i \leq n - 1$) дүгээр элемент бүр нь i дүгээр усан оргилуурын байрлал болох $(x[i], y[i])$, энд $x[i]$ ба $y[i]$ нь тэгш тоонууд.
- Хэрэв байгуулалт хийх боломжтой бол уг функц нь шийдийг олохыг тулд `build` (доороос харна уу) функцийг зөвхөн нэг удаа дуудаж, дараа нь 1 утга буцаах ёстой.
- Үгүй бол уг функц нь `build` функцийг дуудахгүйгээр 0 утга буцаах ёстой.
- Уг функцийг зөвхөн нэг удаа дуудах ёстой.

Таны хэрэгжүүлэлт нь зам байгуулах, вандан сандал байрлуулах боломжийг олж тогтоохын тулд дараах функцийг дуудах болно:

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

- u, v : m урттай хоёр массив, байгуулах замыг илэрхийлнэ. Тэдгээр замууд нь 0 -ээс $m - 1$ хүртэл дугаартай байна. j ($0 \leq j \leq m - 1$) бүрийн хувьд j зам нь $u[j]$ ба $v[j]$ дугаартай усан оргилуурыг холбоно. Зам бүр 2 урттай хэвтээ эсвэл босоо шулууны сегмент байх ёстой. Дурын хоёр ялгаатай зам нь хамгийн ихдээ нэг цэг (усан оргилуур)-ийг агуулж болно. Замуудыг байгуулснаар аль ч хоёр усан оргилуурын хооронд нэг замаар явж очих боломжтой байх ёстой.
- a, b : вандан сандалыг илэрхийлэх m урттай хоёр массив. j ($0 \leq j \leq m - 1$) бүрийн хувьд j дүгээр замд $(a[j], b[j])$ байрлалд байрлуулсан вандан сандал. Ижил байрлалтай хоёр сандал байхгүй. Ижид замд ялгаатай вандан сандал байрлуулсан байх ёсгүй.

Жишээ

Жишээ 1

Дараах дуудалтыг авч үзье:

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

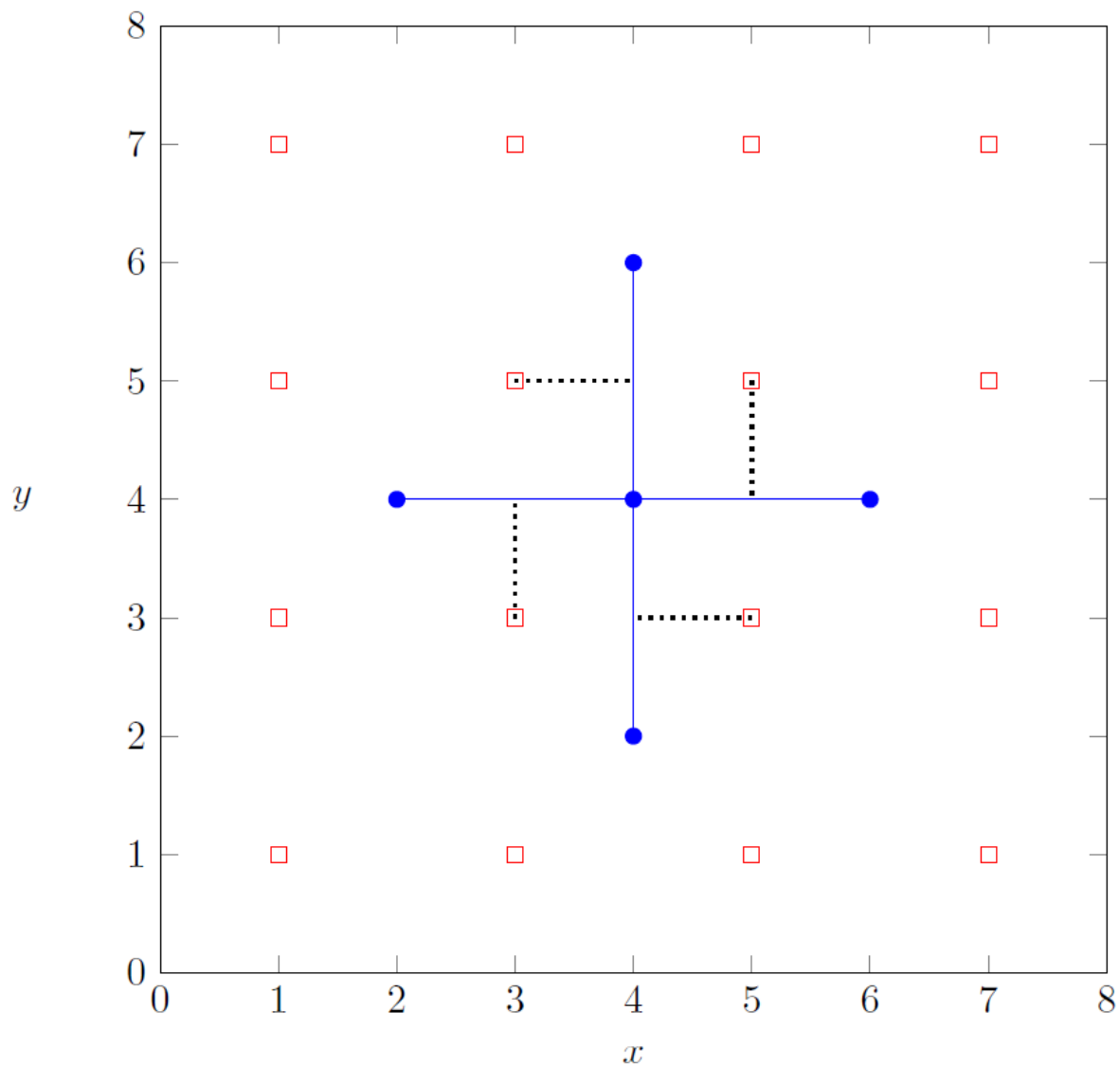
Энэ тохиолдолд 5 усан оргилуур байна:

- 0 дугаар усан оргилуур $(4, 4)$ байрлалд,
- 1 дүгээр усан оргилуур $(4, 6)$ байрлалд,
- 2 дугаар усан оргилуур $(6, 4)$ байрлалд,
- 3 дугаар усан оргилуур $(4, 2)$ байрлалд,
- 4 дүгээр усан оргилуур $(2, 4)$ байрлалд байна.

Зам тус бүр нь хоёр усан оргилуурыг холбосон 4 замыг барьж, зам бүрд вандан сандлыг байрлуулах боломжтой:

Замын дугаар	Замын холбож буй усан оргилуурын дугаар	Хуваарилсан сандлуудын байрлал
0	0, 2	(5, 5)
1	0, 1	(3, 5)
2	3, 0	(5, 3)
3	4, 0	(3, 3)

Энэхүү шийдлийг дараах зургаар илэрхийлж болно:



Энэхүү шийдлийг хийхийн тулд `construct_roads` функцийг дараах байдлаар дуудах ёстой:

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

Энэ дуудалт 1 утга буцаах ёстой.

Энэ тохиолдолд шаардлагад нийцсэн олон шийдэл байгаа бөгөөд эдгээрийг бүгдийг нь зөв гэж үзэх болно. Жишээлбэл, `build([1, 2, 3, 4], [0, 0, 0, 0], [5, 5, 3, 3], [5, 3, 3, 5])` нь мөн зөв дуудалт зөв бөгөөд 1 утга буцаах болно.

Жишээ 2

Дараах дуудалтыг авч үзье:

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

0 усан оргилуур (2,2) байрлалд and 1 усан оргилуур (4,6) байрлалд байна. Шаардлагыг хангасан зам байгуулах боломжгүй учраас `construct_roads` дуудалт нь `build` функцийг

дуудалгүйгээр 0 утга буцаана.

Хязгаарлалт

- $1 \leq n \leq 200\,000$
- $2 \leq x[i], y[i] \leq 200\,000$ (i нь $0 \leq i \leq n - 1$ байхад)
- $x[i]$ ба $y[i]$ нь тэгш бүхэл тоо (i нь $0 \leq i \leq n - 1$ байхад).
- Хоёр усан оргилуур ижил байрлалд байхгүй.

Subtasks

1. (5 оноо) $x[i] = 2$ (i нь $0 \leq i \leq n - 1$ байхад)
2. (10 оноо) $2 \leq x[i] \leq 4$ (i нь $0 \leq i \leq n - 1$ байхад)
3. (15 оноо) $2 \leq x[i] \leq 6$ (i нь $0 \leq i \leq n - 1$ байхад)
4. (20 оноо) Аливаа хоёр усан оргилуурын хооронд нэг замаар зочлох боломжтой нэг боломжтой хамгийн ихдээ нэг зам байгуулах боломжтой байх
5. (20 оноо) 2×2 квадрат үүсгэх дөрвөн усан оргилуур байхгүй.
6. (30 оноо) Нэмэлт хязгаарлалт байхгүй.

Жишээ грейдер

The sample grader reads the input in the following format:

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

The output of the sample grader is in the following format:

- line 1: the return value of `construct_roads`

If the return value of `construct_roads` is 1 and `build(u, v, a, b)` is called, the grader then additionally prints:

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