Tiling Game

Բարսեղը և Չինարը խաղ են խաղում $2N \times 2M$ քառակուսիներից բաղկացած դաշտի վրա։ Տողերը համարակալված են վերևից ներքև 0 -ից մինչև $2N-1$, իսկ սյուները՝ ձախից աջ 0 -ից մինչև $2M-1$ ։ i տողի և j սյան ($0 \le i < 2N$ և $0 \le j < 2M$) վանդակը նշանակում ենք (i, j) -ով։

Բարսեղը Չինարին մեկ առ մեկ տալիս է $N \cdot M$ **բլոկներ**։ Յուրաքանչյուր բլոկ 2×2 քառակուսի է, որը բաղկացած է չորս 1×1 **սալիկներից**։ Բարսեղը բլոկի յուրաքանչյուր սալիկ գունավորել է սև կամ սպիտակ՝ երաշխավորելով, որ **առնվազն մեկ սալիկ սպիտակ է**։

Չինարը պետք է յուրաքանչյուր բլոկը տեղադրի ցանցի վրա **ստանալուց անմիջապես հետո**՝ առանց իմանալու, թե հետագայում ինչ բլոկներ կստանա։ Բլոկները չի կարելի պտտել։ Յուրաքանչյուր բլոկ պետք է ամբողջությամբ տեղադրվի դաշտի ներսում՝ ծածկելով ճիշտ չորս վանդակ։ Ավելին, յուրաքանչյուր բլոկի վերևի ձախ անկյունում գտնվող սալիկը պետք է ծածկի այնպիսի վանդակ, որի տողի և սյան կոորդինատները զույգ են։ Ցանցի յուրաքանչյուր վանդակ կարող է ծածկվել առավելագույնը մեկ բլոկով։

Բարսեղը հաղթում է խաղը, եթե որևէ բլոկի տեղադրումից հետո գոյություն ունի այնպիսի 2×2 քառակուսի, որը ծածկված է չորս սև սալիկներով։ Ֆորմալ, եթե (a, b) , $(a+1, b)$, $(a, b+1)$, $(a+1, b+1)$ վանդակները բոլորը ծածկված են սև սալիկներով որոշակի $0 \le a < 2N-1$ և $0 \le b < 2M-1$ թվերի համար, ապա Բարսեղը հաղթում է։ a և b ինդեքսները պարտադիր չէ, որ զույգ լինեն։

Չինարը կհաղթի, եթե տեղադրի բոլոր $N \cdot M$ բլոկները և Բարսեղը այդ ընթացքում չհաղթի։ Նկատեք, որ $N \cdot M$ բլոկների տեղադրումը ամբողջությամբ կծածկի դաշտը։

Ձեր խնդիրն է մշակել այնպիսի ստրատեգիա, որը կօգնի Չինարին հաղթել խաղը։ Կարելի է ապացուցել, որ տրված սահմանափակումների դեպքում Չինարը միշտ կարող է տեղադրել բլոկները այնպես, որ երաշխավորի հաղթանակը՝ անկախ նրանից, թե հետագայում ինչպիսի բլոկներ է ստանում։

Իրականացման մանրամասներ

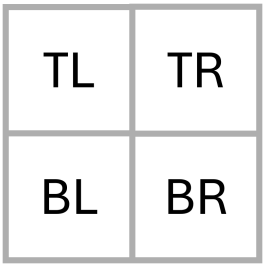
Դուք պետք է իրականացնեք երկու ֆունկցիա.

```
void init(int N, int M)
```

- N : դաշտում տողերի քանակի կեսը։
- M : դաշտում սյուների քանակի կեսը։
- Ֆունկցիան կանչվում է յուրաքանչյուր թեստի համար ճիշտ մեկ անգամ՝ ձեր ծրագրի կատարման սկզբում։

```
std::pair<int, int> receive_block(int TL, int TR, int BL, int BR)
```

- TL , TR , BL , BR : ընթացիկ բլոկի վերևի ձախ, վերևի աջ, ներքևի ձախ և ներքևի աջ սալիկների գույները, համապատասխանաբար, ինչպես ցույց է տրված ստորև բերված նկարում: Յուրաքանչյուր արժեք կամ 0 (սպիտակ) է, կամ 1 (սև):
- Այս ֆունկցիան կանչվում է ճիշտ $N \cdot M$ անգամ մեկ թեստի համար՝ `init` ֆունկցիայի սկզբնական կանչից հետո։



Այս ֆունկցիան պետք է վերադարձնի ամբողջ թվերի զույգ (i, j) , որտեղ i-ն տողն է, իսկ j-ն՝ սյունը, որտեղ պետք է տեղադրվի այս բլոկի վերին ձախ անկյունում գտնվող սալիկը։ Ե՛վ i-ն, և՛ j-ն պետք է զույգ լինեն, և բլոկով ծածկված 2×2 տարածքը չպետք է համընկնի նախկինում տեղադրված որևէ բլոկի հետ։

Եթե `receive_block` ֆունկցիան երբևէ վերադարձնի զույգ, որը չի բավարարում այս պահանջներին, կամ եթե բլոկը տեղադրելուց հետո վանդակների 2×2 քառակուսին ամբողջությամբ ծածկվի սև սալիկներով, գրեյդերը անմիջապես կդադարեցնի ձեր ծրագիրը, և թեստային դեպքի արդյունքը կլինի `Output isn't correct` ՝:

Գրեյդերի վարքագիծը **ադապտիվ չէ**: Սա նշանակում է, որ Բարսեղի կողմից Չինարին տրված բլոկների հաջորդականությունը ֆիքսված է `init` կանչելուց առաջ:

Սահմանափակումներ

Յուրաքանչյուր բլոկի համար, ենթադրենք S ը նրա չորս սալիկներից սև սալիկների քանակն է: Այսինքն՝ $S = TL + TR + BL + BR$:

- $1 \leq N, M \leq 100$
- $0 \leq S \leq 3$ յուրաքանչյուր բլոկի համար։

Ենթախնդիրներ

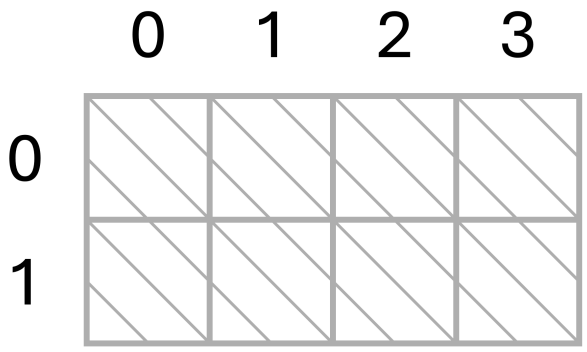
Ենթախնդիր	Գնահատական	Լրացուցիչ սահմանափակումներ
1	6	$S = 1$ յուրաքանչյուր բլոկի համար, և $N = 2$:
2	16	$S = 3$ յուրաքանչյուր բլոկի համար: $N = M$, N ը զույգ թիվ է, և բլոկների չորս հնարավոր գունազարդումներից յուրաքանչյուրը հայտնվում է ճիշտ $\frac{N^2}{4}$ անգամ:
3	10	$S = 1$ յուրաքանչյուր բլոկի համար:
4	29	$S \leq 2$ յուրաքանչյուր բլոկի համար:
5	39	Լրացուցիչ սահմանափակումներ չկան:

Օրինակ

Դիտարկենք մի խաղ, որտեղ $N = 1$ և $M = 2$ է, այսինքն՝ դաշտը ունի 2 տողեր և 4 սյուներ: Սկզբում գրեյդերը կանչում է.

```
init(1, 2)
```

Սկզբում բոլոր բջիջները դատարկ են: Դաշտը այսպիսի տեսք ունի՝

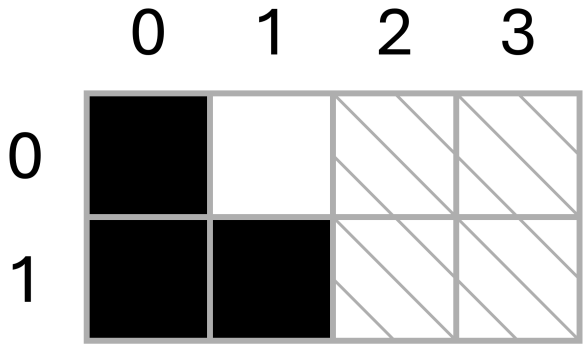


Կան $N \cdot M = 2$ բլոկներ, որոնք պետք է տեղադրել: Ենթադրենք, որ Բարսեղը տալիս է վերին աջ անկյունում մեկ սպիտակ և երեք սևապարունակող բլոկ: Գրեյդերը կանչում է.

```
receive_block(1, 0, 1, 1)
```

Չինարը որոշում է այս բլոկը տեղադրել դաշտի ձախ կողմում՝ վերադարձնելով $(0, 0)$:

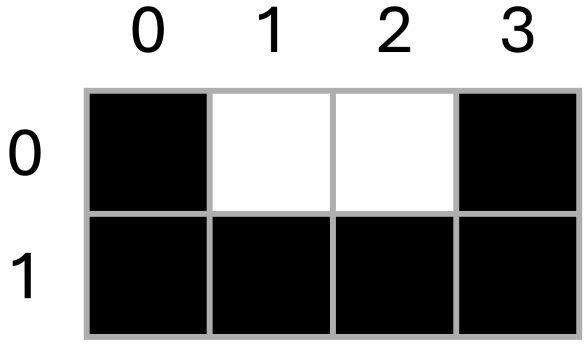
Դաշտն այժմ այսպիսի տեսք ունի՝



Այնուհետև Բարսեղը տալիս է վերին ձախ անկյունում մեկ սպիտակ սալիկ պարունակող և մնացածը սև ներկված բլոկ։

```
receive_block(0, 1, 1, 1)
```

Մնացել է մեկ համապատասխան վանդակ, որը $(0,2)$ն է, ուստի Չինարը վերադարձնում է $(0,2)$: Դաշտն այսպիսին է ստացվում՝



Ոչ մի 2×2 քառակուսի ամբողջությամբ ծածկված չէ սև սալիկներով, ուստի Չինարը հաջողությամբ տեղադրել է բոլոր բլոկները և հաղթել է խաղը։

Նմուշային գրեյդեր

Մուտքի ձևաչափը՝

```
N M
TL[0] TR[0] BL[0] BR[0]
TL[1] TR[1] BL[1] BR[1]
...
TL[NM-1] TR[NM-1] BL[NM-1] BR[NM-1]
```

Ելքի ձևաչափը՝

```
R[0] C[0]
R[1] C[1]
...
R[NM-1] C[NM-1]
```

Այստեղ $R[k]$ և $C[k]$ ամբողջ թվերի զույգ են, որոնք վերադարձվել են `receive_block`k -րդ կանչի կողմից։