Futbol stadioni

Nadyerdo - bu Debresen shahrida joylashgan $N \times N$ o'lchamdagi jadval shaklida ifodalash mumkin bo'lgan kvadrat shaklidagi o'rmon. Jadvalning qatorlari shimoldan janubga tomon 0 dan $N-1$ gacha raqamlangan va ustunlar esa g'arbdan sharqqa tomon 0 dan $N-1$ gacha raqamlangan. Jadvalning r - satrining c - ustunida joylashgan katakchani (r, c) katak deb ataymiz.

O'rmonda har bir katakda **bo'sh** yoki **daraxt** mavjud. O'rmonda kamida bitta katak bo'sh.

Shaharning mashhur sport klubi DVSC o'rmonda yangi futbol stadioni qurishni rejalashtirmoqda. s o‘lchamdagi stadion (bu yerda $s \geq 1$) s ta *bo‘sh* $(r_0, c_0), \ldots, (r_{s-1}, c_{s-1})$ katakchalar to‘plamidir Bu shuni anglatadiki:

- 0 dan $s-1$ gacha bo‘lgan har bir i uchun (r_i, c_i) katak bo‘sh,
- har bir i, j $(0 \leq i < j < s)$ uchun, $r_i \neq r_j$ yoki $c_i \neq c_j$ dan kamida bittasi o'rinli bo'ladi.

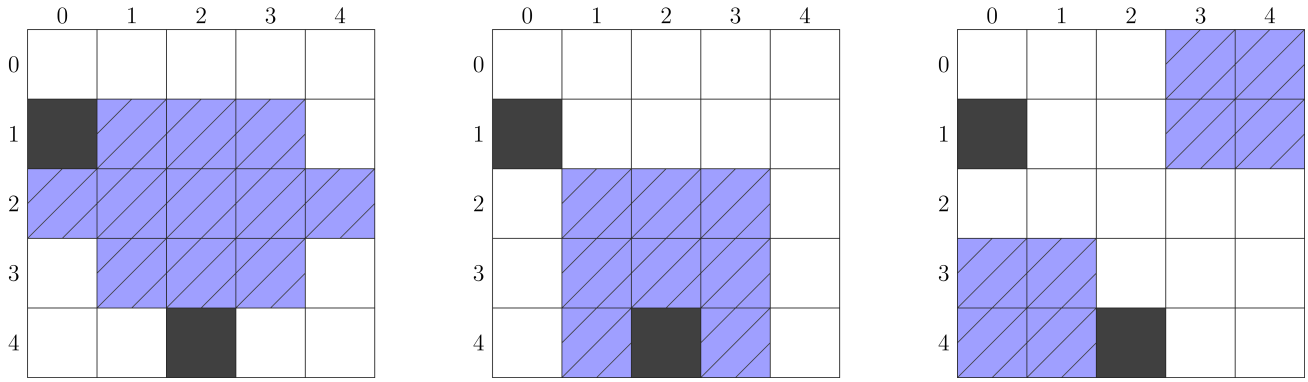
Futbol o'yin to'p yordamida o'ynaladi, u stadion kataklari bo'ylab harakatlanadi. **to'g'ridan-to'g'ri zarba** quyidagi ikkita harakatdan biri sifatida belgilanadi:

- Agar stadion r - satrning a - ustunidan $b (0 \leq r, a, b < N, a \neq b)$ - ustunigacha bo'lgan barcha kataklarni o'z ichiga olsa to'pni (r, a) katagidan (r, b) katagiga to'g'ridan-to'g'ri zarba bilan ko'chirish mumkin. Ya'ni,
  - agar $a < b$ bo'lsa stadion $a \leq k \leq b$ oraliqdagi har bir k uchun (r, k) katakni o'z ichiga olsa,
  - agar $a > b$ bo'lsa stadion $b \leq k \leq a$ oraliqdagi har bir k uchun (r, k) katakni o'z ichiga olsa,
- Agar stadion c - ustunning a - qatoridan $b (0 \leq r, a, b < N, a \neq b)$ - qatorigacha bo'lgan barcha kataklarni o'z ichiga olsa to'pni (a, c) katagidan (b, c) katagiga to'g'ridan-to'g'ri zarba bilan ko'chirish mumkin. Ya'ni,
  - agar $a < b$ bo'lsa stadion $a \leq k \leq b$ oraliqdagi har bir k uchun (k, c) katakni o'z ichiga olsa,
  - agar $a > b$ bo'lsa stadion $b \leq k \leq a$ oraliqdagi har bir k uchun (k, c) katakni o'z ichiga olsa,

To'pni stadionning ixtiyoriy katagidan boshqa bir ixtiyoriy katagiga ko'pi bilan 2 ta to'g'ridan-to'g'ri zarba bilan o'tkazishni imkoni bo'lsa stadion **muntazam** deyiladi. Etibor bering 1 o'lchamdagi har qanday stadion muntazamdir.

Misol uchun, $(1, 0)$ va $(4, 2)$ katakchalarida daraxtlar va boshqa barcha kataklar bo‘sh bo‘lgan $N = 5$ o‘lchamli o‘rmonni ko‘rib chiqaylik. Quyidagi rasmda uchta mumkin bo'lgan stadion

ko'rsatilgan. Daraxtlar bo'lgan kataklar qoraygan va stadiondagi kataklar chiziqli.



Chap tarafdagi stadion muntazam. Biroq o'rtadagi stadion muntazam emas, chunki $(4, 1)$ katakdan $(4, 3)$ katakka o'tish uchun eng kamida 3 ta to'g'ridan-to'g'ri zarba kerak. O'ng tarafdagi stadion ham muntazam emas, chunki to'pni $(3, 0)$ katakdan $(1, 3)$ katakka olib o'tib bo'lmaydi.

Sport klubi iloji boricha kengroq bo'lgan muntazam stadion qurmoqchi. Sizning vazifangiz o'rmonda s o'lchamdagi muntazam stadion mavjud bo'lishi uchun s ning maksimal qiymatini topishdir.

Implementation Details

Siz quyidagi funksiyani implement qilishingiz kerak:

```
int biggest_stadium(int N, int[][] F)
```

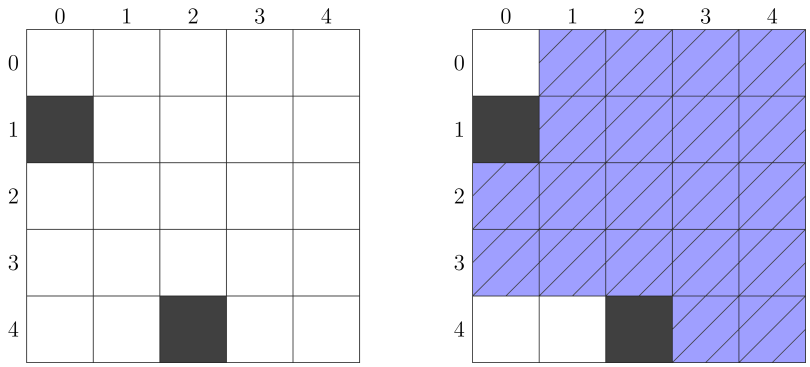
- N: o'rmon o'lchami.
- F: N ta qator va N ta ustundan iborat bo'lgan o'rmonning holatini ifodalaydigan ikki o'lchamli massiv. Har bir $r (0 \leq r < N)$ va $c (0 \leq c < N)$ uchun $F[r][c] = 0$ bo'lsa (r, c) bo'sh katak, va $F[r][c] = 1$ bo'lsa bu katakda daraxt mavjudligini anglatadi.
- Bu funksiya o'rmonda qurish mumkin bo'lgan maksimum o'lchamdagi muntazam stadionning o'lchamini qaytarishi kerak.
- Har bir test case uchun bu funksiya anniq bir marotaba ishlatiladi!

Example

Consider the following call:

```
biggest_stadium(5, [[0, 0, 0, 0, 0],
                    [1, 0, 0, 0, 0],
                    [0, 0, 0, 0, 0],
                    [0, 0, 0, 0, 0],
                    [0, 0, 1, 0, 0]])
```

In this example, the forest is displayed on the left and a regular stadium of size 20 is displayed on the right of the following figure:



Since there is no regular stadium of size 21 or greater, the procedure should return 20.

Constraints

- $1 \leq N \leq 2\,000$
- $0 \leq F[i][j] \leq 1$ (for each i and j such that $0 \leq i < N$ and $0 \leq j < N$)
- There is at least one empty cell in the forest. In other words, $F[i][j] = 0$ for some $0 \leq i < N$ and $0 \leq j < N$.

Subtasks

1. (6 points) There is at most one cell containing a tree.
2. (8 points) $N \leq 3$
3. (22 points) $N \leq 7$
4. (18 points) $N \leq 30$
5. (16 points) $N \leq 500$
6. (30 points) No additional constraints.

In each subtask, you can obtain 25% of the subtask score if your program judges correctly whether the set consisting of *all* the empty cells is a regular stadium.

More precisely, for each test case in which the set consisting of all the empty cells is a regular stadium, your solution:

- gets full points if it returns the correct answer (which is the size of the set consisting of all the empty cells).
- gets 0 points otherwise.

For each test case in which the set consisting of all the empty cells is *not* a regular stadium, your solution:

- gets full points if it returns the correct answer.
- gets 0 points if it returns the size of the set consisting of all the empty cells.
- gets 25% of the points if it returns any other value.

The score for each subtask is the minimum of the points for the test cases in the subtask.

Sample Grader

The sample grader reads the input in the following format:

- line 1: N
- line $2 + i$ ($0 \leq i < N$): $F[i][0]$ $F[i][1]$ $\ldots$ $F[i][N-1]$

The sample grader prints your answer in the following format:

- line 1: the return value of `biggest_stadium`