

Футбольний стадіон

Нагьердо - це ліс квадратної форми, який знаходиться в місті Дебрецен. Він може бути представлений як поле з клітинками. Розміри поля - $N \times N$. Столпчики поля пронумеровані від 0 до $N - 1$ зліва направо, а рядки пронумеровані від 0 до $N - 1$ зверху вниз. Клітинку, яка знаходиться у рядку r та столпчику c , будемо називати клітинкою (r, c) .

У лісі кожна клітинка або **порожня**, або містить **дерево**. Принаймні одна клітинка в лісі порожня.

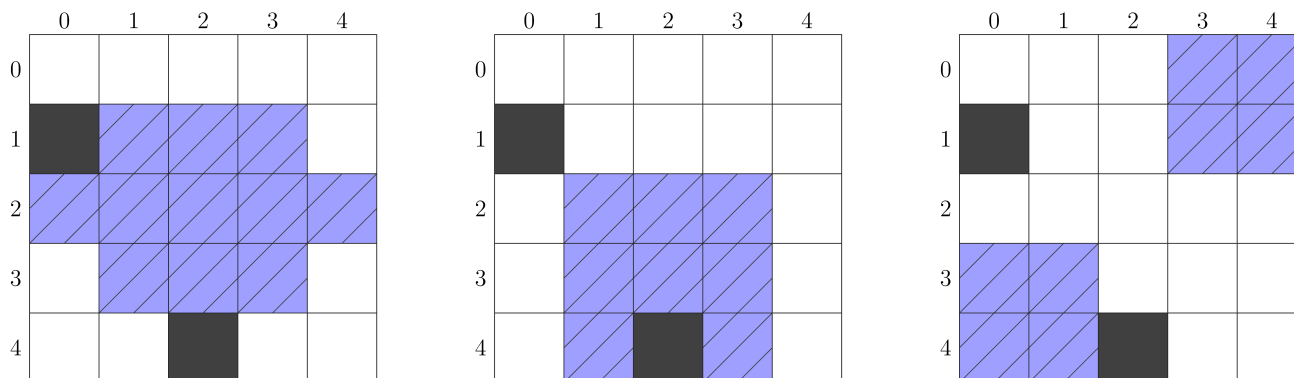
DVSC, відомий спортивний клуб в місті, планує побудувати новий футбольний стадіон в лісі. Стадіон розміру s (де $s \geq 1$) - це набір s різних порожніх клітинок $(r_0, c_0), \dots, (r_{s-1}, c_{s-1})$. Тобто, для кожного i від 0 до $s - 1$ включно клітинка (r_i, c_i) порожня, а також для кожного j такого, що $i < j < s$, виконується $r_i \neq r_j$ або $c_i \neq c_j$.

У футбол грають за допомогою м'яча, який пересувають по клітинках стадіону. **Прямий удар** визначається як одна з наступних двох дій:

- Пересування м'яча з клітинки (r, a) в клітинку (r, b) ($0 \leq r, a, b < N, a \neq b$), якщо стадіон містить всі клітинки між клітинками (r, a) та (r, b) у рядку r . Формально,
 - якщо $a < b$, то стадіон має містити клітинку (r, k) для всіх k таких, що $a \leq k \leq b$,
 - якщо $a > b$, то стадіон має містити клітинку (r, k) для всіх k таких, що $b \leq k \leq a$.
- Пересування м'яча з клітинки (a, c) в клітинку (b, c) ($0 \leq c, a, b < N, a \neq b$), якщо стадіон містить всі клітинки між клітинками (a, c) та (b, c) у столпчику c . Формально,
 - якщо $a < b$, то стадіон має містити клітинку (k, c) для всіх k таких, що $a \leq k \leq b$,
 - якщо $a > b$, то стадіон має містити клітинку (k, c) для всіх k таких, що $b \leq k \leq a$.

Стадіон називається **правильним**, якщо можливо пересунути м'яч з будь-якої клітинки стадіону у будь-яку іншу клітинку стадіону, використовуючи не більше ніж 2 прямих удари. Зверніть увагу, що стадіон з розміром 1 є правильним.

Наприклад, розглянемо ліс розміру $N = 5$, в якому клітинки $(1, 0)$ та $(4, 2)$ містять дерева, а всі інші клітинки порожні. Малюнки нижче показують можливі стадіони. Клітинки з деревами пофарбовані в чорний, а клітинки стадіону заштриховані.



Стадіон зліва є правильним. Проте, стадіон посередині таким не є оскільки принаймні 3 прямих удари потрібні, щоб перемістити м'яч з клітинки (4, 1) в (4, 3). Стадіон справа також не є правильним через те, що неможливо перемістити м'яч з клітинки (3, 0) в (1, 3), використовуючи прямі удари.

Спортивний клуб хоче побудувати правильний стадіон максимально можливого розміру. Ваша задача полягає в тому, щоб знайти максимальне можливе значення s таке, що існує правильний стадіон розміром s .

Деталі імплементації

Вам потрібно імплементувати наступну функцію:

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

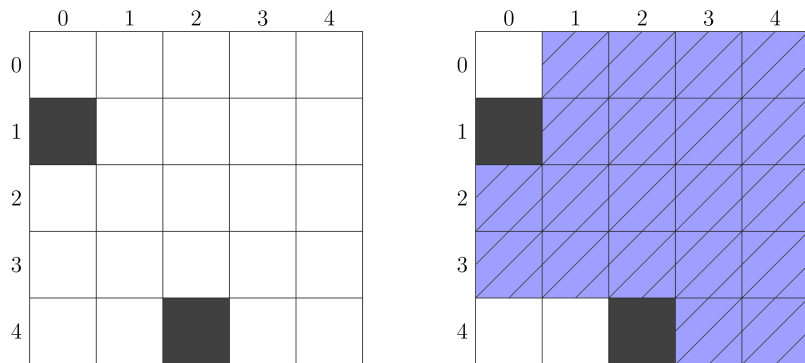
- N : розмір лісу.
- F : масив довжини N , який містить масиви довжини N , які описують клітинки лісу. Для кожного r та c таких, що $0 \leq r < N$ та $0 \leq c < N$, $F[r][c] = 0$ означає, що клітинка (r, c) порожня, а $F[r][c] = 1$ означає, що ця клітинка містить дерево.
- Ця функція має повернути максимальний розмір правильного стадіону, який можна побудувати в лісі.
- Ця функція викликається рівно раз.

Приклад

Розглянемо наступний виклик:

```
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]])
```

У цьому прикладі ліс зображено ліворуч, а правильний стадіон розміром 20 - праворуч:



Оскільки немає правильного стадіону розміру принаймні 21, то функція має повернути 20.

Обмеження

- $1 \leq N \leq 2000$
- $0 \leq F[i][j] \leq 1$ (для кожного i та j таких, що $0 \leq i < N$ та $0 \leq j < N$)
- Є принаймні одна порожня клітинка. Іншими словами, $F[i][j] = 0$ для деяких $0 \leq i < N$ та $0 \leq j < N$.

Підзадачі

1. (6 балів) Є не більше однієї клітинки, яка містить дерево.
2. (8 балів) $N \leq 3$
3. (22 бали) $N \leq 7$
4. (18 балів) $N \leq 30$
5. (16 балів) $N \leq 500$
6. (30 балів) Без додаткових обмежень.

У кожній підзадачі ви можете отримати часткові бали, якщо ваша програма визначає правильно, чи підмножина з усіх порожніх клітинок є правильним стадіоном.

Точніше, для кожного тесту, у якому підмножина з усіх порожніх клітинок є правильним стадіоном, ваше рішення:

- отримає всі бали, якщо воно поверне правильну відповідь (яка також є розміром підмножини з усіх порожніх клітинок).
- отримує 0 балів в іншому випадку.

Для кожного тесту, у якому підмножина з усіх порожніх клітинок *не* є правильним стадіоном, ваше рішення:

- отримає всі бали, якщо воно поверне правильну відповідь.
- отримає 0 балів, якщо воно поверне розмір підмножини з усіх порожніх клітинок.
- отримає 25% балів, якщо воно поверне будь-яке інше число.

Кількість балів за підзадачу обчислюється як мінімальна кількість балів серед усіх тестів підзадачі.

Приклад градера

Градер зчитує дані у наступному форматі:

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

Градер виводить дані у наступному форматі:

- рядок 1: число, яке повернула функція `biggest_stadium`