



Estádio de Futebol

Nagyerdő é uma floresta de formato quadrado localizada na cidade de Debrecen, que pode ser modelada como uma grade de $N \times N$ células. As linhas da grade são numeradas de 0 até $N - 1$ do norte para o sul, e as colunas são numeradas de 0 to $N - 1$ do oeste para o leste. Vamos nos referir à célula localizada na linha r e coluna c da grade como célula (r, c) .

Na floresta, cada célula é ou **vazia** ou contém uma **árvore**. Pelo menos uma célula na floresta é vazia.

DVSC, o famoso clube de esportes da cidade, está planejando construir um novo estádio de futebol na floresta. Um estádio de tamanho s (onde $s \geq 1$) é um conjunto de s células *vazias distintas* $(r_0, c_0), \dots, (r_{s-1}, c_{s-1})$. Formalmente, isto significa:

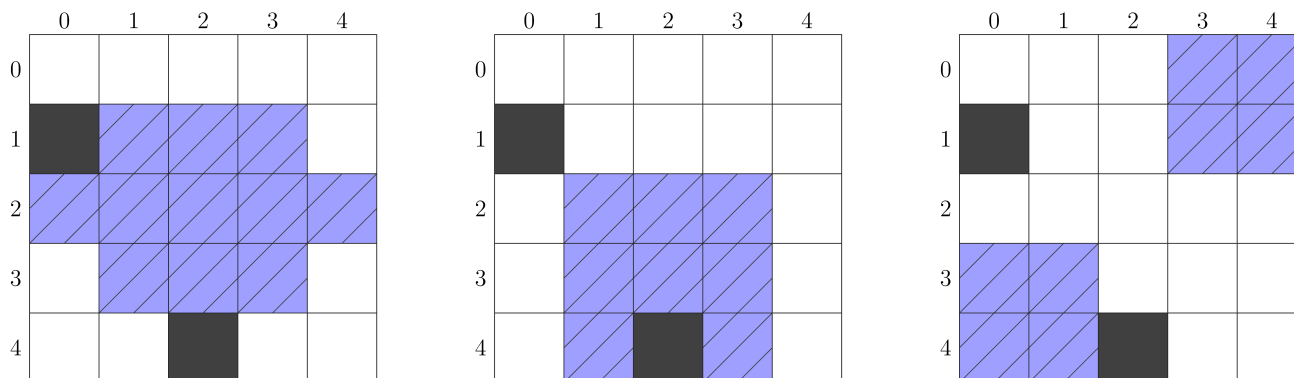
- para cada i de 0 até $s - 1$, inclusive, a célula (r_i, c_i) é vazia,
- para cada i, j tal que $0 \leq i < j < s$, pelo menos uma entre as restrições $r_i \neq r_j$ e $c_i \neq c_j$ é verdadeira.

Futebol é jogado usando uma bola que se move pelas células do estádio. Um **chute reto** é definido como uma das seguintes duas ações:

- Mover a bola da célula (r, a) para a célula (r, b) ($0 \leq r, a, b < N, a \neq b$), onde o estádio contém *todas* as células entre a célula (r, a) e a célula (r, b) na linha r . Formalmente,
 - se $a < b$ então o estádio deveria conter a célula (r, k) para cada k tal que $a \leq k \leq b$,
 - se $a > b$ então o estádio deveria conter a célula (r, k) para cada k tal que $b \leq k \leq a$.
- Mover a bola da célula (a, c) para a célula (b, c) ($0 \leq c, a, b < N, a \neq b$), onde o estádio contém *todas* as células entre (a, c) e (b, c) na coluna c . Formalmente,
 - se $a < b$ então o estádio deveria conter a célula (k, c) para cada k tal que $a \leq k \leq b$,
 - se $a > b$ então o estádio deveria conter a célula (k, c) para cada k tal que $b \leq k \leq a$.

Um estádio é **regular** se é possível mover a bola de qualquer célula contida pelo estádio para qualquer outra célula contida pelo estádio com no máximo 2 chutes diretos. Note que qualquer estádio de tamanho 1 é regular.

Por exemplo, considere uma floresta de tamanho $N = 5$, com as células $(1, 0)$ e $(4, 2)$ contendo árvores e cada outra célula sendo vazia. A figura abaixo mostra três estádios possíveis. Células com árvores são escuras, e células contidas pelo estádio são listradas.



O estádio à esquerda é regular. No entanto, o estádio no meio não é regular, porque pelo menos 3 chutes diretos são necessários para mover a bola da célula (4,1) para (4,3). O estádio da direita também não é regular, porque é impossível mover a bola da célula (3,0) para (1,3) usando chutes diretos.

O clube de esportes deseja construir um estádio regular que seja o maior possível. Sua tarefa é encontrar o valor máximo de s tal que exista um estádio regular de tamanho s na floresta.

Detalhes de Implementação

Você deve implementar o seguinte procedimento.

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

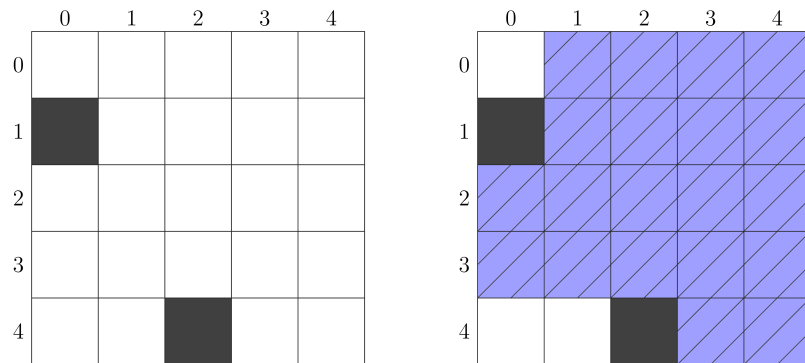
- N : o tamanho da floresta.
- F : um vetor de comprimento N contendo vetores de comprimento N , descrevendo células na floresta. Para cada r e c tal que $0 \leq r < N$ e $0 \leq c < N$, $F[r][c] = 0$ indica que a célula (r, c) é vazia, e $F[r][c] = 1$ indica que a célula contém uma árvore.
- Este procedimento deve retornar o tamanho máximo de um estádio regular que pode ser construído na floresta.
- Este procedimento é chamado exatamente uma vez para cada caso de teste.

Exemplo

Considere a seguinte chamada:

```
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]])
```

Neste exemplo, a floresta é mostrada à esquerda e um estádio regular de tamanho 20 é mostrado à direita da seguinte figura:



Como não há estádio regular de tamanho 21 ou maior, o procedimento deve retornar 20.

Restrições

- $1 \leq N \leq 2000$
- $0 \leq F[i][j] \leq 1$ (para cada i e j tal que $0 \leq i < N$ e $0 \leq j < N$)
- Há pelo menos uma célula vazia na floresta. Em outras palavras, $F[i][j] = 0$ para algum $0 \leq i < N$ e $0 \leq j < N$.

Subtasks

1. (6 pontos) Há no máximo uma célula contendo uma árvore.
2. (8 pontos) $N \leq 3$
3. (22 pontos) $N \leq 7$
4. (18 pontos) $N \leq 30$
5. (16 pontos) $N \leq 500$
6. (30 pontos) Nenhuma restrição adicional.

Em cada subtarefa, você pode obter 25% da pontuação da subtarefa se o seu programa julga corretamente se o conjunto formado por *todas* as células vazias é um estádio regular.

Mais precisamente, para cada caso de teste em que o conjunto formado por todas as células vazias é um estádio regular, sua solução:

- recebe a pontuação total se ela retorna a resposta correta (que é o tamanho do conjunto formado por todas as células vazias).
- recebe 0 pontos caso contrário.

Para cada caso de teste em que o conjunto formado por todas as células vazias *não* é um estádio regular, sua solução:

- recebe a pontuação total se ela retorna a resposta correta
- recebe 0 pontos se ela retorna o comprimento do conjunto formado por todas as células vazias.
- recebe 25% dos pontos se retorna qualquer outro valor.

A pontuação de cada subtarefa é o mínimo dos pontos para os casos de teste na subtarefa.

Corretor Exemplo

O corretor exemplo lê a entrada no seguinte formato:

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

O corretor exemplo imprime sua resposta no seguinte formato:

- linha 1: o valor de retorno de `biggest_stadium`