



Stadio calcistico

Nagyerdő è una foresta quadrata $N \times N$ nella città di Debrecen, con righe e colonne numerate da 0 a $N - 1$. Nella foresta, ogni cella è **vuota** oppure contiene un **albero**. Almeno una cella è vuota.

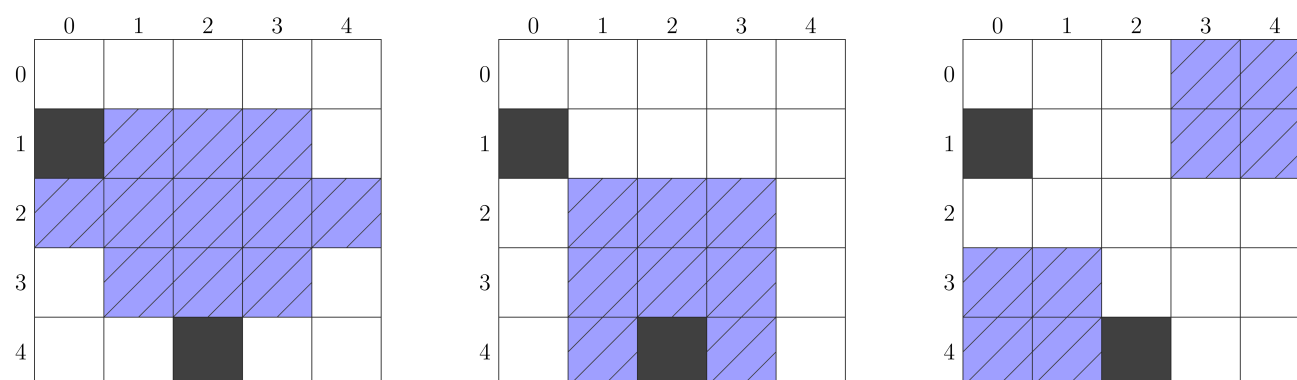
Il famoso centro sportivo DVSC vuole costruire uno stadio calcistico nella foresta. Uno stadio di dimensione $s \geq 1$ è un insieme di s celle *distinte e vuote* $(r_0, c_0), \dots, (r_{s-1}, c_{s-1})$.

A calcio si gioca con una palla che può occupare una delle celle dello stadio. Un **passaggio** è una delle due azioni seguenti:

- Muovere la palla orizzontalmente, dalla cella (r, a) alla cella (r, b) ($a \neq b$), dove lo stadio contiene *tutte* le celle comprese tra (r, a) e (r, b) .
- Muovere la palla verticalmente, dalla cella (a, c) alla cella (b, c) ($a \neq b$), dove lo stadio contiene *tutte* le celle comprese tra (a, c) e (b, c) .

Uno stadio si dice **regolare** se, per ogni coppia di celle dello stadio (distinte), è possibile muovere la palla dalla prima alla seconda con al più 2 passaggi. In particolare, ogni stadio di dimensione 1 è regolare.

Per esempio, considera una foresta di dimensione $N = 5$, con due alberi nelle celle $(1, 0)$ e $(4, 2)$ (in nero), e tutte le altre celle vuote. La figura qui sotto mostra tre possibili stadi (blu a strisce).



Lo stadio a sinistra è regolare, mentre quello al centro non lo è, poiché almeno 3 passaggi sono necessari per muovere la palla da $(4, 1)$ a $(4, 3)$. Anche lo stadio a destra non è regolare, perché è impossibile muovere la palla da $(3, 0)$ a $(1, 3)$ con dei passaggi.

Il centro sportivo vuole costruire uno stadio regolare il più grande possibile: trova il massimo valore s per cui esiste uno stadio regolare di dimensione s !

Note di implementazione

Devi implementare la seguente funzione:

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

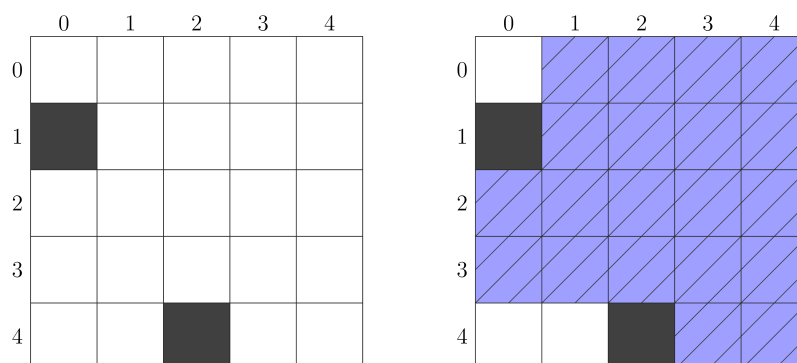
- N : la lunghezza del lato della foresta.
- F : una matrice $N \times N$ che descrive la foresta: $F[r][c] = 0$ se la cella (r, c) è vuota, e $F[r][c] = 1$ se contiene un albero.
- Questa funzione deve restituire la massima dimensione di uno stadio regolare che può essere costruito nella foresta.
- Questa funzione viene chiamata esattamente una volta per ciascun testcase.

Esempio

Considera la seguente chiamata:

```
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]])
```

La foresta di questo esempio è mostrata a sinistra e ammette uno stadio regolare di dimensione 20 , mostrato a destra:



Si può constatare che non esiste uno stadio regolare di dimensione almeno 21, per cui la funzione deve restituire 20.

Assunzioni

- $1 \leq N \leq 2000$.
- $0 \leq F[i][j] \leq 1$ per ogni $0 \leq i, j < N$.
- La foresta contiene almeno una cella vuota, ovvero F contiene almeno uno 0.

Subtask

1. (6 punti) C'è al massimo un albero.
2. (8 punti) $N \leq 3$.
3. (22 punti) $N \leq 7$.
4. (18 punti) $N \leq 30$.
5. (16 punti) $N \leq 500$.
6. (30 punti) Nessuna limitazione aggiuntiva.

In ogni subtask è possibile ottenere un punteggio parziale del **25%**, nel caso in cui il programma determini se il sottoinsieme di *tutte* le celle vuote è uno stadio regolare.

Più precisamente, per ogni testcase in cui il sottoinsieme di tutte le celle vuote è uno stadio regolare, la soluzione ottiene:

- punteggio pieno se restituisce la risposta corretta, ovvero il numero di celle vuote;
- altrimenti, 0 punti.

Invece, per ogni testcase in cui il sottoinsieme di tutte le celle vuote *non* è uno stadio regolare, la tua soluzione ottiene:

- punteggio pieno se restituisce la risposta corretta;
- 0 punti se restituisce il numero di celle vuote;
- il 25% dei punti del subtask se restituisce ogni altro valore.

Il punteggio di un subtask è il minimo dei punteggi ottenuti dai testcase che contiene.

Grader di esempio

Il grader di esempio legge l'input nel seguente formato:

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

Il grader di esempio stampa la risposta nel seguente formato:

- riga 1: il valore di ritorno di `biggest_stadium`