



## Fudbalski stadion

Nagyerdő je šuma kvadratnog oblika koja se nalazi pored Beograda, koju možemo predstaviti kao rešetka sa  $N \times N$  polja. Redovi su numerisani od 0 do  $N - 1$  od sjevera prema jugu, a kolone su numerisane od 0 do  $N - 1$  od zapada prema istoku. Polje koje se nalazi u redu  $r$  i koloni  $c$  rešetke nazivamo kao polje  $(r, c)$ .

U šumi, svako od polja je ili **prazno** ili sadrži **drvo**. Barem jedno polje je prazno.

DVSC, poznat sportski klub iz grada, planira da izgradi novi fudbalski stadion u Nagyerdő-skoj šumi. Stadion veličine  $s$  (gde je  $s \geq 1$ ) je skup od  $s$  *različitih praznih* polja  $(r_0, c_0), \dots, (r_{s-1}, c_{s-1})$ . To znači da

- za svako  $i$  od 0 do  $s - 1$ , polje  $(r_i, c_i)$  je prazno
- za svako  $j$  takvo da  $i < j < s$ , važi  $r_i \neq r_j$  ili  $c_i \neq c_j$  (ili oboje).

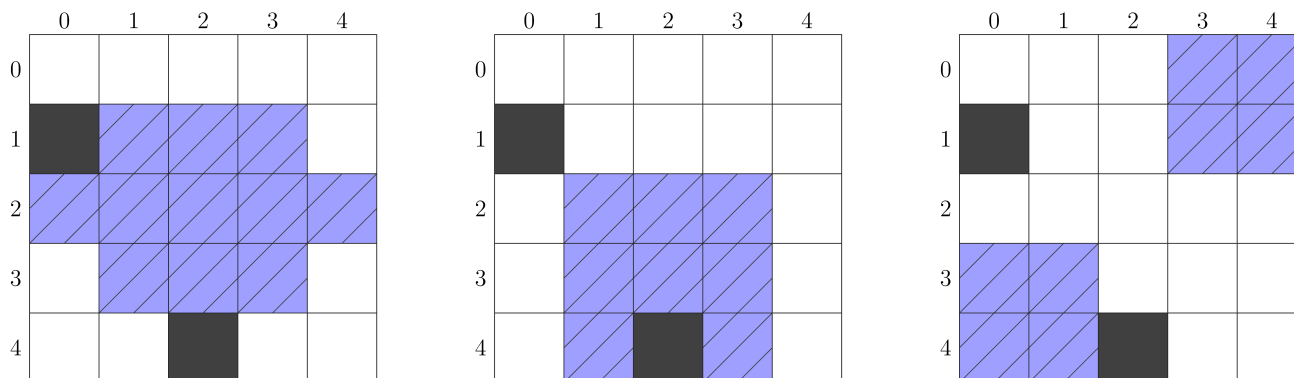
Fudbal se igra koristeći loptu koja može da se pomjera po poljima stadiona.

**Špic-šut** je definisan kao jedna od dvije navedene akcije:

- Pomjeranje lopte od polja  $(r, a)$  u polje  $(r, b)$  ( $0 \leq r, a, b < N, a \neq b$ ), gdje stadion sadrži *sva* polja između početnog polja  $(r, a)$  i krajnjeg polja  $(r, b)$  u redu  $r$ . Tačnije,
  - ako je  $a < b$  onda stadion mora da sadrži  $(r, k)$  za svako  $k$  tako da važi  $a \leq k \leq b$ ,
  - ako je  $a > b$  onda stadion mora da sadrži  $(r, k)$  za svako  $k$  tako da važi  $b \leq k \leq a$ .
- Pomjeranje lopte od polja  $(a, c)$  u polje  $(b, c)$  ( $0 \leq c, a, b < N, a \neq b$ ), gdje stadion sadrži *sva* polja između početnog polja  $(a, c)$  i krajnjeg polja  $(b, c)$  u koloni  $c$ . Tačnije,
  - ako je  $a < b$  onda stadion mora da sadrži  $(k, c)$  za svako  $k$  tako da važi  $a \leq k \leq b$ ,
  - ako je  $a > b$  onda stadion mora da sadrži  $(k, c)$  za svako  $k$  tako da važi  $b \leq k \leq a$ .

Stadion je **pravilan** ako je moguće da se lopta pomjeri iz bilo kog polja koje je sadržano u stadionu do bilo kog polja koje je sadržano u stadionu u **najviše 2 špic-šuta**. Primjetimo da je bilo koji stadion veličine 1 pravilan.

Na primjer, posmatrajmo šumu veličine  $N = 5$ , sa poljima  $(1, 0)$  i  $(4, 2)$  koje sadrže drveće i svim ostalim poljima koja su prazna. Na slici ispod možemo da vidimo tri stadiona. Polja sa drvećem su zatamnjena, a polja koja su dio stadiona su šrafirana (i plava).



Stadion lijevo je pravilan. Stadion u sredini nije pravilan, zato što su najmanje 3 špic-šuta potrebna da se lopta pomjeri sa polja (4, 1) na polje (4, 3). Stadion desno takođe nije pravilan, zato što je nemoguće pomjeriti loptu sa polja (3, 0) na polje (1, 3) špic-šutevima.

DVSC želi da napravi stadion tako da bude što je moguće veći, a takođe i da bude pravilan. Vaš zadatak je da nađete najveću vrijednost  $s$  takvu da se pravilan stadion veličine  $s$  može napraviti u Nagyerdő-skoj šumi.

## Detalji implementacije

Treba da implementirate sljedeću funkciju.

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

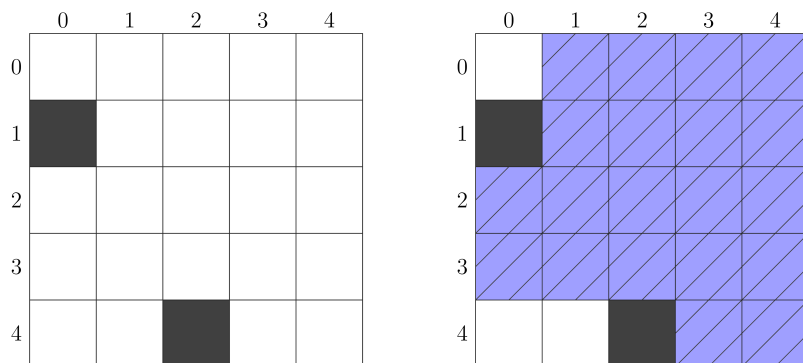
- $N$ : veličina šume.
- $F$ : niz dužine  $N$  koji sadrži nizove dužine  $N$ , koji opisuju polja u šumi. Za svako  $r$  i  $c$  tako da je  $0 \leq r < N$  i  $0 \leq c < N$ ,  $F[r][c] = 0$  znači da je polje  $(r, c)$  prazno, a  $F[r][c] = 1$  znači da sadrži drvo.
- Ova funkcija treba da vrati maksimalnu veličinu pravilnog stadiona koji može da se napravi u šumi.
- Ova funkcija će biti pozvana tačno jednom za svaki test primer.

## Primjer

Posmatrajmo sledeći poziv funkcije:

```
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]])
```

U ovom primjeru, šuma je prikazana na slici lijevo i pravilan stadion veličine 20 je prikazan na slici desno:



Obzirom na to da ne postoji pravilan stadion sa veličinom 21 i više, funkcija treba da vrati 20.

## Ograničenja

- $1 \leq N \leq 2000$
- $0 \leq F[i][j] \leq 1$  (za svako  $i$  i  $j$  tako da važi  $0 \leq i < N$  i  $0 \leq j < N$ )
- Postoji bar jedno prazno polje u šumi. Drugim riječima,  $F[i][j] = 0$  za neko  $0 \leq i < N$  i  $0 \leq j < N$ .

## Podzadaci

1. (6 bodova) Postoji najviše jedno polje koje sadrži drvo.
2. (8 bodova)  $N \leq 3$
3. (22 bodova)  $N \leq 7$
4. (18 bodova)  $N \leq 30$
5. (16 bodova)  $N \leq 500$
6. (30 bodova) Bez dodatnih ograničenja.

U svakom podzadatku, možete da osvojite 25% bodova za taj podzadatak ukoliko vaš program tačno odredi da li skup koji se sastoji od *svih* praznih polja predstavlja pravilan stadion.

Prezicnije, za svaki testni primjer u kome je skup svih praznih polja pravilan stadion, vaše rješenje:

- dobija sve bodove ukoliko vrati tačan odgovor (koji je veličina skupa koji sadrži sva prazna polja).
- u suprotnom, dobija 0 bodova.

Za svaki testni primjer u kome skup svih praznih polja *nije* pravilan stadion, vaše rješenje:

- dobija sve bodove ukoliko vrati tačan odgovor.
- dobija 0 bodova ukoliko vrati veličinu skupa koji se sastoji od svih praznih polja.
- dobija 25% poena ukoliko vrati bilo koji drugu vrijednost.

Broj bodova osvojen u svakom podzadatku je minimum poena koji su osvojili test primjeri u tom podzadatku.

## Primjer gradera

Primer gradera učitava ulaz u sljedećem formatu:

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

Grader ispisuje vaše rešenje u sljedećem formatu:

- linija 1: povratna vrijednost funkcije `biggest_stadium`