



Nogometni stadion

Nagyerdő je gozd kvadratne oblike v mestu Debrecen, ki ga lahko modeliramo kot mrežo celic $N \times N$. Vrstice mreže so oštevilčene od 0 do $N - 1$ (od severa proti jugu) in stolpci so oštevilčeni od 0 do $N - 1$ (od zahoda proti vzhodu). Celico, ki se nahaja v r -ti vrstici in c -tem stolpcu mreže, imenujemo celica (r, c) .

V gozdu je vsaka celica ali **prazna** ali pa vsebuje kakšno **drevo**. Vsaj ena celica v gozdu je prazna.

DVSC, znani mestni športni klub, namerava v gozdu zgraditi nov nogometni stadion. Stadion velikosti s (kjer je $s \geq 1$) je množica s različnih praznih celic $(r_0, c_0), \dots, (r_{s-1}, c_{s-1})$ (nobena celica se ne pojavi več kot enkrat v množici). Formalno to pomeni:

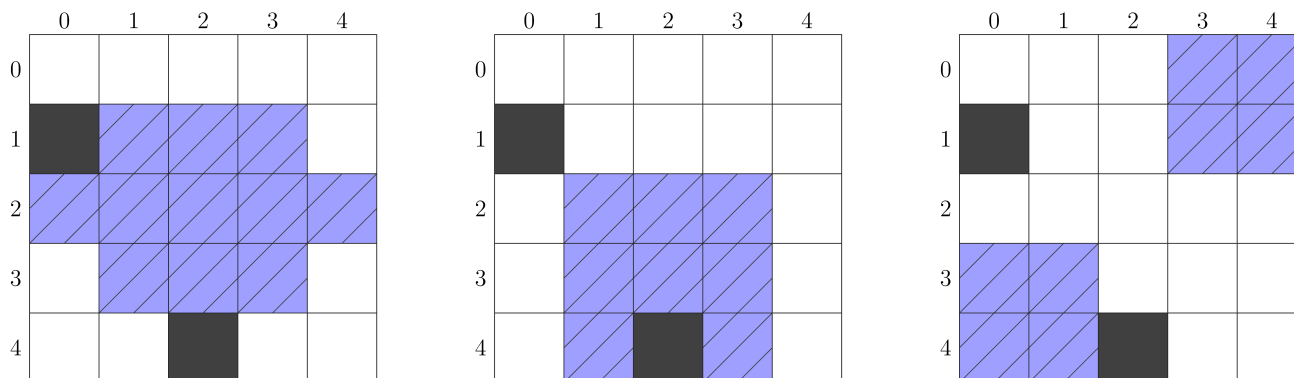
- za vsak i od 0 do vključno $s - 1$ je celica (r_i, c_i) prazna,
- za vsak i, j , kjer je $0 \leq i < j < s$, velja $r_i \neq r_j$ ali $c_i \neq c_j$ ali oboje.

Nogomet se igra z žogo, ki se premika po celicah stadiona. Za **direktni udarec** velja eno od naslednjih dveh dejanj:

- Žogo premaknemo iz celice (r, a) v celico (r, b) ($0 \leq r, a, b < N, a \neq b$), pri čemer stadion vsebuje vse celice med celicama (r, a) in (r, b) v vrstici r . Formalno,
 - če $a < b$, potem mora stadion vsebovati celice (r, k) , za vsak k , kjer je $a \leq k \leq b$,
 - če $a > b$, potem mora stadion vsebovati celice (r, k) za vsak k , kjer je $b \leq k \leq a$.
- Žogo premaknemo iz celice (a, c) v celico (b, c) ($0 \leq c, a, b < N, a \neq b$), pri čemer stadion vsebuje vse celice med celicama (a, c) in (b, c) v stolpcu c . Formalno,
 - če $a < b$, potem mora stadion vsebovati celice (k, c) za vsak k , kjer je $a \leq k \leq b$,
 - če $a > b$, potem mora stadion vsebovati celice (k, c) za vsak k , kjer je $b \leq k \leq a$.

Stadion je **regularen**, če je žogo možno premakniti iz katere koli celice stadiona v katero koli drugo celico stadiona, z največ 2 direktnimima udarcema. Upoštevaj, da je vsak stadion velikosti 1 pravilen.

Na primer, upoštevajmo gozd velikosti $N = 5$, v katerem sta v celicah $(1, 0)$ in $(4, 2)$ drevesa, vse ostale celice so pa prazne. Spodnja slika prikazuje tri možne stadione. Celice z drevesi so zatemnjene, celice, ki predstavljajo stadion, pa so črtaste.



Stadion na levi je regularen. Stadion na sredini ni regularen, saj so za premik žoge iz celice (4, 1) do (4, 3) potrebni vsaj 3 direktni udarci. Stadion na desni prav tako ni regularen, ker je premik iz celice (3, 0) v (1, 3) nemogoč.

Športni klub želi zgraditi čim večji regularen stadion. Tvoja naloga je poiskati največjo vrednost s , da bo v gozdu obstajal regularen stadion velikosti s .

Podrobnosti implementacije

Implementirajte naslednjo funkcijo.

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

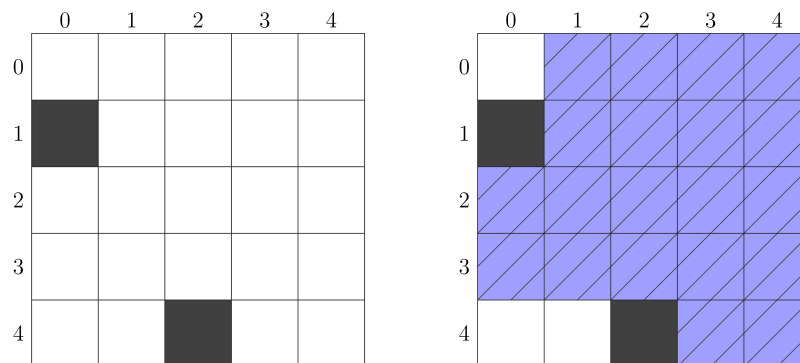
- N : velikost gozda.
- F : polje dolžine N , ki vsebuje polja dolžine N , ki opisujejo celice v gozdu. Za vsak r in c , kjer $0 \leq r < N$ in $0 \leq c < N$, $F[r][c] = 0$, kar pomeni, da je celica (r, c) prazna; sicer $F[r][c] = 1$, kar pomeni, da vsebuje drevo.
- Funkcija mora vrniti največjo velikost regularnega stadiona, ki ga je mogoče zgraditi v gozdu.
- Funkcija se za vsak testni primer pokliče natanko enkrat.

Primer

Upoštevajte naslednji klic:

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

V tem primeru je na levi strani prikazan gozd, na desni strani pa je prikazan pravilen stadion velikosti 20:



Ker ni regularnega stadiona velikosti 21 (niti večjega), mora funkcija vrniti 20.

Omejitve

- $1 \leq N \leq 2000$
- $0 \leq F[i][j] \leq 1$ (za vsak i in j tako, da $0 \leq i < N$ in $0 \leq j < N$)
- V gozdu je vsaj ena prazna celica. Z drugimi besedami, $F[i][j] = 0$ za nekatere $0 \leq i < N$ in $0 \leq j < N$.

Podnaloge

1. (6 točk) Obstaja največ ena celica, ki vsebuje drevo.
2. (8 točk) $N \leq 3$
3. (22 točk) $N \leq 7$
4. (18 točk) $N \leq 30$
5. (16 točk) $N \leq 500$
6. (30 točk) Brez dodatnih omejitev.

Pri vsaki podnalogi lahko dobite 25% točk, če vaš program pravilno oceni, ali je podmnožica, ki jo sestavljajo vse prazne celice, regularen stadion.

Natančneje, za vsak testni primer, v katerem je podmnožica, sestavljena iz vseh praznih celic, regularen stadion, vaša rešitev:

- dobi polno število točk, če vrne pravilen odgovor (ki je velikost podmnožice, sestavljene iz vseh praznih celic).
- v nasprotnem primeru dobi 0 točk.

Za vsak testni primer, v katerem podmnožica, sestavljena iz vseh praznih celic, *ni* regularen stadion, vaša rešitev:

- dobi polno število točk, če je odgovor pravilen.
- dobi 0 točk, če vrne velikost podmnožice, sestavljene iz vseh praznih celic.
- dobi 25 % točk, če vrne katero koli drugo vrednost.

Rezultat za vsako podnalogo je najmanjša vrednost doseženih točk vseh testnih primerov podnaloge.

Vzorčni ocenjevalnik

Vzorčni ocenjevalnik bere vhod naslednje oblike:

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

Vzorčni ocenjevalnik izpiše odgovor v naslednji obliki:

- vrstica 1: vrnjena vrednost funkcije `biggest_stadium`