O-timberland

Ovaj zadatak je iz kategorije “Complete Search” (kompletno pretraživanje).

Algoritam:

- Ulazni podaci su u obliku upita, sa kombinovanim tekstualnim i brojčanim podacima u istom redu.
- Potrebno je zadati broj **Qi** rastaviti na delioce i spremiti parove čiji je on proizvod.
- Za sve tako pripremljene parove koji sada predstavljaju stranice pravougaonika treba proveriti ima li stabala u takvom pravougaoniku. (za sve moguće načine da se ceo pravougaonik postavi u polje)
- Ako ima bar jedan takav pravougaonik odgovor je “**yes**” inače “**no**”.

BRUTE FORCE pristup

Dogadjaji CUT i GROW se mogu rešiti u O(1) vremenu, pa ih nećemo razmatrati.

VOTE je dogadjaj je ključni u ovom zadatku.
Samo rastavljanje broja **Qi** na parove je složenosto $O(N^2)$, što neće ključno uticati na vreme ako in pripremimo pre kompletnog pretraživanja s obzirom na N <= 100.
Ali za svaki ovaj dogadjaj O(MaxVote), za svaki par stranica O(BrojParova), potrebno je za svaki položaj tog pravougaonika $O(N^2)$ proveriti ima li stablo unutar njega $O(N^2)$.
Memorijska složenost algoritma je $O(N^2)$
Vremenska složenost algoritma je MaxVote * ProsBrojParova * $O(N^4)$
(MaxVote * ProsBrojParova je oko 10000, sto je ukupno približno oko 10^{12})

BRUTE Force pristip donosi i do 60 bodova, u zavisnosti od oprimizacije koda.

Gde se može ostvariti razlika:

1. Rastavljanje na činioce do korena od N, (jedna strana je manja od korena a druga veca)
2. Vertikalne orijentacije se mogu dobiti zamenom stranica iz horizontalne (ili obratno)
3. Isključiti sve provere za pravougaonike kojima je jedna stranica > N
4. Čim se nadje jedno stablo unutar pravougaonika preći na sledeću proveru.
5. Čim se nadje jedan odg. pravougaonik, preći na sledeći upit

Pristup KUMULETIVNIM TABELAMA

Priprema parova stranica pravougaonika je razmatrana u BRUTE pristupu.

Poznato da kumulativne tabele imaju vremensku složenost log(N) po dimenziji i za postavljenje i za pronalaženje. Prednost u odnosu na BRUTE FORCE bi trebala da postoji. To se empirijski očekuje.
O(logN*logN) što je približno 50, za N=100 u odnosu na $N^{2} = 10000$. Medjutim:

Operacija	Vreme za KUMULATIVNE	Vreme za BRUTE FORCE
Postavljanje	50.000 * 50	50.000 * 1
Nalaženje	365 * 50 * 4	365 * 10000
Ukupno	2573000	3700000

Ako se uzme u obzir da je pri uzimanju i postavljanju podataka u KT, potrebno dve naredbe dodele u odnosu na jednu kod BRUTE FORCE pristupa, vidimo da KUMULATIVNE tabele u ovom zadatku ne donose nikakvu prednost, čak su inferiorne (KT=515 : BRUTE =370)

Ovo je veoma interesantna stvar, a glavni razlog je mala dimenzija N. (samo 100)

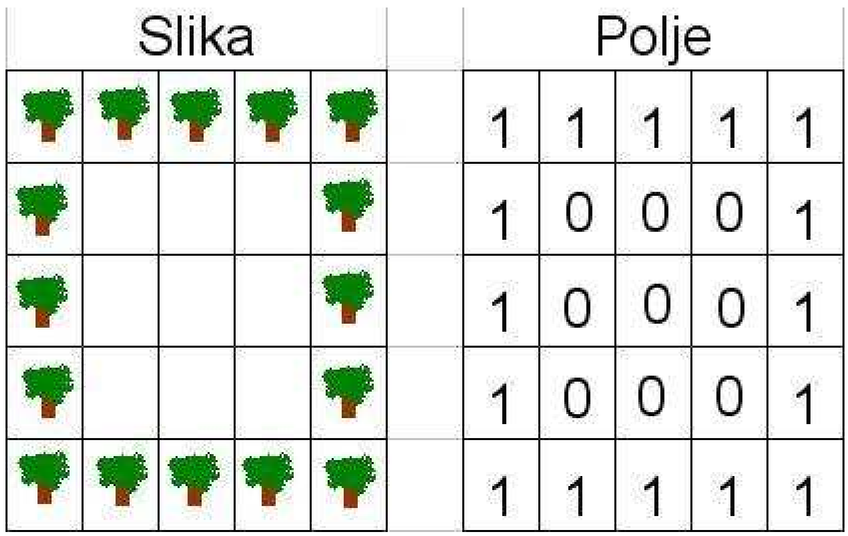
Kumulativne oko 50 poena. (u PASCAL-u možda nešto više u C ili C++)

Pristup DINAMIČKIM PROGRAMIRANJEM

Jedina razlika koja se može ostvariti jeste u delu pronalaženja stabala u nekom pravougaoniku.

BRUTE FORCE	$O(N^2)$
KUMULATIVNE TABELE	$O(\log^2 N)$
DINAMIČKI	$O(1)$

Ovo spušta ukupnu složenost za N^2 tj 10000 pa sa ukupnih 10^8 može da se uklopi u ograničenja u zadatku.



Slika 1 Slika 2

Dinamički možemo pripremiti matricu Z iz koje je kasnije moguće direktno proveriti da li u nekoj pravougaonoj oblasti postoji drvo. Matricu gradimo počev od donjeg levog ugla tako da svaki član matrice predstavlja broj stabala unutar pravougaone oblasti (1,1) – donji levi ugao i (i,j)- gornji desni ugao.

Zbir[(1,1)..(i,j)]

0	5	7	9	11	16
0	4	5	6	7	11
0	3	4	5	6	9
0	2	3	4	5	7
0	1	2	3	4	5
0	0	0	0	0	0

Slika 3

O(1)

0	5	7	9	11	16
0	4	5	6	7	11
0	3	4	5	6	9
0	2	3	4	5	7
0	1	2	3	4	5
0	0	0	0	0	0

Slika 4

Svaki član matrice nastaje kao Z[i,j] = Polje[i,j] + Z[i, j-1] + Z[i-1, j] - Z[i-1, j-1]

Nakon formiranja matrice Z, moguće je direktno saznati ima li u nekoj pravougaonoj oblasti stabla ili ne.
Na slici 4 se vidi da je

Z[4,4] + Z[1,1] = Z[1,4] + Z[4,1] a to je moguće jedino ako u oblasti [(2,2) .. (4,4)] ne postoji nijedno drvo. Dokaz ostavljamo vama.
Analogno ovome, moguća je provera za bilo koju pravougaoni oblast.

NAPOMENA, matricu zbir je potrebno odole i sa leva prosiriti za nultu kolonu i nulti red, u kojima trebaju biti nule.

Autor:
Duško Obradović