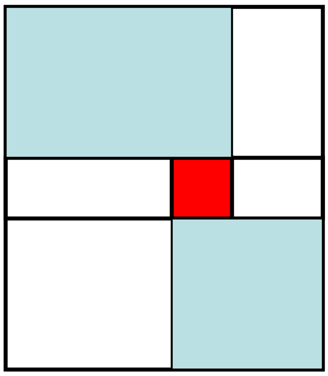
Matrice

Problema se poate rezolva folosind divide et impera. Proprietatea de care ne folosim e aceea că daca scoatem linii şi coloane din matricea iniţială, matricea rămasă e total monotonă.

Solutia 1



Se caută elementul maxim de pe linia din mijloc a matricii curente, folosind N query-uri. Apoi se apelează recursiv pentru submatricile necesare. După cum se observă în figura alăturată, din cauza proprietăţii enunţate mai sus, căutarea nu trebuie reluată decât pe porţiunile albastre. Această soluţie foloseşte O(N log N) interogări, întrucât la fiecare pas se elimină aproximativ jumatate din matrice.

Solutia 2

După cum s-a văzut în prima soluţie, acolo s-au eliminat doar linii şi coloane consecutive din matricea iniţială. Soluţia optima foloseste tot divide et impera, dar exploatează într-un mod mai efficient proprietăţile matricelor total monotone. Pentru o matrice dată, se selectează iniţial doar liniile pare si se rezolvă recursiv pentru submatricea compusă de acestea. Odată având poziţiile elementelor maxime pentru aceste linii pare, se pot calcula cu uşurinţă elementele maxime si pentru poziţiile impare folosind N interogări.
Mai rămâne de stabilit cum se rezolvă cazul în care matricea are N linii şi 2*N coloane. Este evident faptul că cel puţin N dintre coloanele matricei nu conţin maximul nici unei linii (deoarece avem doar N linii). Se pot identifica folosind O(N) interogări N coloane ce nu conţin maxime. Odată găsite, aceste coloane sunt eliminate, iar algoritmul continuă pentru matricea pătratică rămasă. Iată pseudocodul pentru găsirea a N coloane fără maxim:

A’ <- A; k <- 1;
while A’ are mai mult de N coloane **do**
    if A’(k, k)≥A’(k, k+1) **and** k < n **then**
        k++;
    else
        if A’(k, k)≥A’(k, k+1) **and** k=n **then**
            Stergem coloana N+1 din A’
        else
            if A’(k, k) **<** A’(k, k+1) **then**
                Stergem coloane k din A’
                k--

Se poate demonstra usor că algoritmul descris foloseste cel mult 10*N interogari. La fiecare nivel din recursivitate, trebuie să efectuăm o normalizare (maxim 4*N interogari) şi să aflăm poziţiile minimelor pe liniile impare (încă N interogări), deci calculând recurenta T(N) = 5*N+T(N/2) = 10*N. Din cele 10*N interogari se scad 4*N (deoarece la primul pas nu se efectuează normalizarea), deci rămân maxim 6*N interogări. Este evident faptul că trebuie să ţinem valorile interogate si local, pentru a nu repeta unele interogări. În practică, numărul de interogări este mai mic, deoarece unele valori (mai ales în momentul aflării maximului pe liniile impare) sunt deja interogate.