xor - Descrierea soluţiei

Filip Cristian Buruiană, Universitatea Politehnică Bucureşti

Soluţia trivială calculează răspunsul pentru fiecare submatrice în parte iterând efectiv elementele submatricei. O astfel de abordare nu se încadrează în timp şi obtine 20 de puncte.

Pentru obţinerea a 40 de puncte, putem realiza o preprocesare folosind sume parţiale pe matrice. Notând cu $S_{i,j}$ suma elementelor din submatricea de coordonate (0, 0, i, j), cu i şi j între 2000, putem calcula răspunsul pentru fiecare query în timp constant. Astfel, răspunsul pentru submatricea de coordonate (L_1, C_1, L_2, C_2) va fi $S_{L2,C2} - S_{L2,C1-1} - S_{L1-1,C2} + S_{L1-1,C1-1}$.

Soluţia care obţine 100 de puncte calculează suma unei submatrice în timp logaritmic în dimensiunea unei laturi. Se observă că putem calcula eficient $S_{i,j}$ fără a parcurge efectiv toată matricea. Calculăm răspunsul independent pentru fiecare bit în parte şi apoi combinăm rezultatele. Astfel, pentru bitul k trebuie să determinăm câte perechi (x, y), cu x între 0 şi i şi y între 0 şi j, au al k-lea bit diferit (fie este 0 în prima coordonată şi 1 în a doua, fie invers). Dacă notăm cu NR numărul de perechi în care bitul k este diferit, atunci vom aduna la soluţie 2^k * NR. În general, trebuie să mai determinăm de câte ori apare bitul 0 pe poziţia k în numerele de la 0 la i. Notând acest număr cu $count_i$, obţinem relaţia:
$NR = count_i*(j-count_j) + (i-count_i)*count_j$.
Observăm că bitul k se schimbă alternativ din 2^k în 2^k atunci când parcurgem numerele naturale începând cu 0. Astfel, putem calcula $count_i$ în O(1), folosind câteva observaţii matematice.