

Problema FFT2D

Andrei-Costin Constantinescu, Oxford University

50 puncte - $O(T^2)$

Observatie 1. Drumul dintre oricare 2 noduri, daca exista, este unic.

Sortam nodurile negre de „jos in sus” (in functie de nivel).

Consideram dinamica $dp[node]$ = cate drumuri care pornesc din nodul **node** exista care se termina pe un nod de pe nivelul $F - 1$ si care nu trec printr-un alt nod negru. Observam ca numarul total de lanturi care incep din nodul **node** aflat pe nivelul h si se termina pe ultimul nivel, fara alte restrictii, este dat de 2^{F-h-1} . Ramane ca din acest numar sa scadem drumurile ce trec prin macar inca un nod negru. Pentru a numara aceste drumuri, iteram un nod j la care i poate ajunge si scadem din $dp[node]$ valoarea lui $dp[j]$, semnificand faptul ca scadem din numarul total de drumuri pe toate acelea ce il au pe j ca nod negru cel mai apropiat de nivelul $F - 1$, bazandu-ne pe observatia ca fiecare astfel de drum va fi contorizat o singura data.

100 puncte - $O(F \cdot F \cdot T)$, $O(F \cdot T)$

Observatie 2. Exista drum de la nodul $i(h_1, x_1)$ la nodul $j(h_2, x_2)$ (unde $h_1 \leq h_2$) daca si numai daca x_1 si x_2 difera in scrierile binare doar intre bitii $F - h_2 - 1$ si $F - h_1 - 1$.

Pornind de la solutia anterior prezentata si Observatia 2, vom procesa nodurile date in grupe pe nivel, cate o grupa pentru fiecare nivel, si iterand nivelul nodului j la care se face referire la finalul primei solutii. Pentru o grupa si un nivel fixate, dorim sa efectuam scaderile respective mentionate in prima solutie. Pentru a face acest lucru eficient, vom precalcuila o masca binara pentru fiecare nod (h, x) de pe nivelul nodului j si una pentru fiecare nod (h', x') de pe nivelul nodului i , astfel incat **mask[h, x] = x in care subsecventa dintre bitii $F - h' - 1$ si $F - h - 1$ a fost setata pe 0, iar mask[h', x'] = x' in care subsecventa dintre bitii $F - h' - 1$ si $F - h - 1$ a fost setata pe 0**. Avand aceste 2 precaluclari, vom scadea din $dp[x, h]$ doar acele valori $dp[x', h']$ pentru care **mask[h, x] = mask[h', x']**. Acest lucru se poate face eficient, in timp proportional cu numarul de noduri de pe nivelul h folosind hash-uri, map-uri sau un sir sortat si cautare binara, diversele implementari ducand la complexitati diferite, toate insa de 100 de puncte.

Nota Daca va intrebati de ce problema contine in nume acronimul FFT, ei bine, el vine de la Fast Fourier Transform, un algoritm matematic folosit pentru calculul rapid al produsului a doua polinoame. Implementarea iterativa a acestui algoritm da nastere implicita la un graf, numit **Butterfly Network**, care este chiar cel vizat de problema de fata.