

IHOT 2021-22 Runda Finala / Prosoft@NT 2022 - Solutii

Comisia IHOT

Martie 2022

1 Problema Pepi and Friends

AUTOR: ANDREI BOGDAN NICULA

1.1 Descrierea soluției

Dificultate: Grea

Pentru a rezolva aceasta problema mai intai trebuie sa rezolvam o problema mult mai usoara: cazul in care numarul de oameni imbracati in albastru este egal cu numarul de oameni imbracati in galben.

Ce va trebui sa facem pentru fiecare ID este sa numaram numarul de biti de 1 din acel numar. Daca un student are un ID cu numar impar de biti de 1 il vom numi ID impar, iar altfel il vom numi ID par. Se poate observa cu usurinta faptul ca un student cu ID par poate vorbi doar cu un student cu ID impar (dar nu cu toti) si viceversa. Apoi putem colora studentii dupa paritatea ID-ului. Studentii cu ID impar pot fi colorati in albastru si cei cu ID par pot fi colorati cu galben.

Acum sa ne intoarcem la problema originala unde numarul de oameni colorati intr-o culoare nu are voie sa fie egal cu numarul de oameni colorati cu cealalta culoare. Putem sa construim solutia folosind solutia descrisa anterior.

Pentru fiecare ID putem lua reprezentarea binara si sa o impartim intr-un numar egal de bucket-uri de lungime $\lceil \sqrt{N} \rceil$ incepand de la primul bit (vom umple ultimul bucket cu zerouri pentru a ajunge la lungimea ceruta). Pentru fiecare ID vom verifica daca are un bucket cu toti bitii setati cu valoarea 0. Daca ID-ul are un asemenea bucket atunci ii vom schimba culoarea. Este usor de observat ca pentru oricare student care isi va schimba culoarea doar $\lceil \sqrt{N} \rceil$ studenti vor avea aceeasi culoare ca el, ceea ce respecta cerinta problemei. Numarul de schimbari de culori va fi mereu un numar impar, deci suntem siguri ca numarul de oameni imbracati cu albastru este diferit de numarul de oameni imbracati cu galben.

2 Problema Tabel

AUTOR: BOGDAN IOAN POPA

2.1 Descrierea soluției

Dificultate: Grea

Pentru a putea determina valoarea dintr-o celulă necunoscută aceasta va trebui să fie singura necunoscută pe linia ei sau pe coloana ei.

Vom considera următorul graf bipartit: În partea stângă se află N noduri, fiecare reprezentând câte o linie, în partea dreaptă se află M noduri, fiecare reprezentând câte o coloană. Dacă o celulă (i, j) este necunoscută, atunci vom trage muchie de cost $B[i][j]$ de la nodul linie i la nodul coloană j . Se observă că vom putea rezolva tabelul cu cost 0 dacă graful este un arbore.

Ne interesează să aflăm un arbore parțial de cost cât mai mare, astfel încât costul pe care trebuie să-l plătim ca să ajungem la el să fie cât mai mic. Acest lucru se poate face cu algoritmul lui Kruskal pentru 80 de puncte sau cu algoritmul lui Prim pentru 100 de puncte.

3 Problema Giga Xor

AUTOR: IOAN POPESCU

3.1 Descrierea soluției

Dificultate: Medie

Pentru a calcula suma xor a unui interval $[a, b]$, trebuie să calculăm pentru fiecare bit dacă apare de un număr par sau un număr impar, dacă un bit apare de un număr impar de ori, atunci apare în suma xor a numerelor din intervalul $[a, b]$. Pentru a determina de câte ori apare un anumit bit, putem de exemplu să considerăm că bitul cu valoarea 2^k , este 0 pentru numerele din intervalul $[0, 2^k - 1]$, apoi 1 pe intervalul $[2^k, 2^k + 1 - 1]$, șamd. Astfel, trebuie să vedem câte intervale care au bitul 1 avem, și să avem grijă la intervalele care intersectează în capete intervalul $[a, b]$.

Acum, observația principală, era aceea că suma xor a 4 numere consecutive începând cu un multiplu de 4 este 0. Adică $0 \text{ xor } 1 \text{ xor } 2 \text{ xor } 3 = 0$, $4 \text{ xor } 5 \text{ xor } 6 \text{ xor } 7 = 0$, șamd.

Este ușor de observat că putem extinde brut intervalul $[a, b]$ cu 3 poziții la stânga sau la dreapta.

4 Problema Metin 2

AUTOR: IOAN POPESCU

4.1 Descrierea soluției

Dificultate: Grea

O solutie partiala, intersecteaza iterativ patrulater. Complexitate de $O(N^2)$.

Pentru fiecare patrulater ABCD, consideram 4 vectori, care sunt determinati de laturile orientate in asa fel incat partea din stanga a vectorilor sa fie spre interiorul patrulaterului. Apoi, aplicam algoritmul de intersectie de semiplane, si determinam aria.

Complexitate $O(N \log N)$.

5 Problema Bicolored Matrix

AUTOR: STEFAN DASCALESCU

5.1 Descrierea soluției

Dificultate: Usoara

Inainte de a rezolva oricare dintre cerinte, va trebui sa aflam pentru fiecare coloana folosind sumele partiale intervalele in care putem avea patrate egale cu 0, respectiv cu 1.

Pentru a rezolva prima cerinta, vom afla raspunsul separat pentru fiecare coloana si vom inmulti raspunsurile individuale. Pentru a face asta, numaram numarul de puncte care pot reprezenta o granita dintre patratele completate cu 0 si patratele completate cu 1.

Pentru cea de-a doua cerinta, vom avea dinamica $dp[i][j] =$ numarul de solutii distincte daca am ajuns la coloana i si granita de la coloana i este pe linia j . Pentru a respecta conditia referitoare la ordinea crescatoare, $dp[i][j]$ va fi suma tuturor valorilor de forma $dp[i-1][p]$, unde $p \leq j$.

Pentru a optimiza aceasta solutie, se pot folosi sume partiale pentru a memora sumele mentionate anterior.

Complexitatea algoritmului este $O(n * m)$.

6 Problema No Division Allowed

AUTOR: STEFAN DASCALESCU

6.1 Descrierea soluției

Dificultate: Medie

În primul rând, se poate observa faptul că pentru a afla câte numere nu sunt divizibile cu niciunul din numerele din intervalul $[2, x]$, este de ajuns să verificăm doar câte numere nu sunt divizibile cu niciunul din numerele prime din acel interval.

Deoarece sunt doar cel mult 8 numere prime în acest interval (valoarea maximă a lui x este 20) Folosirea principiului includerii și excluderii este necesară pentru a rezolva problema, deoarece pentru a afla numărul de numere care nu sunt divizibile cu niciunul dintre aceste numere, este mai simplu să aflăm câte numere sunt divizibile cu cel puțin unul dintre acele numere și să scădem acest rezultat din numărul total de numere din intervalul $[a, b]$.

Pentru a calcula în mod eficient soluția, va trebui să ne generăm toate măștile pe biti ale numerelor prime și să aflăm separat pentru fiecare dintre ele care este răspunsul.

Complexitatea finală va fi $O(2^{NrPrime} * |b|)$, unde $|b|$ este lungimea numărului b .

7 Problema Essential

AUTORI: PROF. ADRIAN PANAETE

7.1 Descrierea soluției

Dificultate: Medie

Pentru a rezolva problema, se poate observa faptul că o muchie (x, y) este necesară dacă drumul minim de la x la y este egal cu costul muchiei (x, y) și acesta este unic (nu există nicio altă secvență de muchii de la x la y care să aibă același cost sau mai mic).

Pentru a verifica ușor această condiție, vom rula algoritmul Roy-Floyd pentru a afla toate drumurile minime între oricare două noduri.

Complexitatea algoritmului este $O(n^3)$.