Descrierea Soluțiilor
Concursul Național “InfoPro”, Runda 3
Grupa C1

1 Problema Cort

Propunător: prof. Flavius Boian C.N. ”Spiru Haret”,
Targu-Jiu

Pentru a rezolva **prima cerinta** a problemei, observam ca putem numara cate linii au cel putin o valoare de 1, sa zicem ca acest numar este X, pentru ca putem sa o fixam pe aceea pe prima coloana a fiecarei linii. Raspunsul acestei cerinte este X, pentru ca putem aseza toate liniile care au cel putin o valoare de 1, in jumatatea superioara a matricei, iar restul liniilor (ce au toate valorile nule), in jumatatea inferioara.
Pentru a rezolva **cea de-a doua cerinta**, vom numara pentru fiecare linie, cate valori de 1 sunt. Vom tine un vector in care vom numara pentru fiecare valoare de la 1 la 1000, cate linii au atatea elemente de 1. Stim ca un dreptunghi are lungime si latime. Tot ce trebuie sa facem acum, este sa parcurgem toate valorile de la 1000 la 1, asftel vom fixa latimea dreptunghiului. Lungimea va fi numarul de valori care sunt mai mari sau egale decat latimea. Putem calcula usor acest lucru intr-o variabila P; la ficare pas dupa ce calculam aria dreptunghiului (produsul dintre lungime si latime), vom aduna la P, valoarea vectorului din pozitia ’latime’.

2 Problema IZI Stack

Propunător: Ușurelu Florian,
Universitatea din București + Universitatea Constantin Brâncuși

Observatia principala a acestei probleme este ca operatiile de tip "adauga tuturor elementelor din stiva de la pozitia x la pozitia y valoarea add", nu trebuie facuta efectiv, pentru ca problema ne cere sa afisam la fiecare pas doar valoarea din varful stivei. In rezolvarea triviala, am fi parcurs toate elementele de la pozitiile x, la y, dar tot ce trebuie sa facem este sa tinem minte pozitiile ce trebuie modificate, respectiv valoarea care va fi adaugata, pentru a lasa pe mai tarziu efectuarea operatiilor. Pentru aceasta, vom tine un vector V, unde $V[i]$ reprezinta valoarea cu care trebuie adunate toate elementele din stiva, de la pozitia i (numarand de jos in sus), pana la baza. De altfel, vom tine si o stiva S cu elementele adaugate sau eliminate, fara a tine cont de operatia de tip 1. Daca toate operatiile de **tip 1** ar fi avut x-ul egal cu 1, solutia noastra ar fi avut sens.

Problema se pune atunci cand x-ul nu este 1, adica atunci cand nu trebuie sa adunam in toate elementele de la y pana la baza, mai concret, trebuie sa oprim aceasta propagare completa. Daca vrem sa adaugam valoarea add, solutia naturala ar fi sa scadem add, acolo unde nu avem voie sa propagam. Putem face acest lucru, adunand in vectorul V la pozitia $x-1$, valoarea $-add$ ($V[x-1]- = add$). Aceasta abordare seamana ca cea din Jmenul lui Mars.

Tot ce mai trebuie sa facem acum este ca inainte sa eliminam un element din varful stivei, sa afisam valoarea din varful stivei S, la care sa adunam $V[marimeaStivei]$ (add-urile ce trebuie propagate). Mai avem o ultima problema de rezolvat. Stim ca in $V[marimeaStivei]$ se afla suma valorilor de trebuie propagate in jos, iar noi vrem sa tinem in continuare cont de aceasta. Trebuie sa adunam la $V[marimeaStivei-1]$, valoarea din $V[marimeaStivei]$, pentru a sti in continuare ce trebuie sa propagam; nu uitam, de altfel, sa "resetam" $V[marimeaStivei]$ pentru operatiile viitoare, adica sa-i atribuim valoarea 0.

Complexitatea acestei solutii este $O(N)$. Mentionam ca se putea lua punctaj maxim cu o solutie $O(NlogN)$ folosind arbori de intervale sau arbori indexati binar.

3 Problema Tomi

Propunător: Ușurelu Florian,
Universitatea din București + Universitatea Constantin Brâncuși

Pentru cei initiati intr-ale operatiilor pe biti, problema se transpune pe scurt in: "gasiti suma OR minima, astfel incat sa fie masca pentru cel putin K elemente din cele date" (un element este in masca, daca toti bitii sai de 1 sunt in acea masca, de exemplu 1101 este masca pentru 0100 sau 1001).

Solutia intuitiva, care obtine 57 de puncte, este sa fixam aceasta masca. Putem sa iteram prin toate valorile de la 1 pana la $sumaOR$ a elementelor de la 1 la 60, verificand la fiecare iteratie daca masca fixata contine cel putin K elemente. Prima masca gasita, fiind cea mai mica, este raspunsul cerintei.

Solutia ce obtine punctaj maxim se bazeaza pe cautarea raspunsului, bit cu bit. Consideram masca initiala ca fiind un numar ce se poate reprezenta pe 60 de biti, iar toti acesti biti fiind initial egali cu 1. Ideea principala in aceasta abordare este ca aportul bitului de pe pozitia $i+1$ de la dreapta la stanga, este mai mare decat aportul tuturor celorlalti biti din dreapta sa. Adica:
$2^i > 2^{i-1} + 2^{1-2} + ... + 2^1 + 2^0$.

Ce vom face, va fi sa eliminam bitul de pe pozitia $i+1$ pentru a vedea daca vom avea cel putin K elemente care se afla in masca. Adica scadem din masca 2^i. Daca dupa ce am eliminat acest bit, masca inca este conforma, trecem mai departe si aplicam aceeasi strategie si pentru restul bitilor din dreapta. Daca dupa ce eliminam acest bit, masca nu mai este conforma, inseamna ca acest bit era unul important, deci il vom pastra.

Complexitatea este $O(logMasca * N)$, adica facem 60 de iterari prin cele N elemente.

Echipa care a pregătit setul de probleme pentru această rundă a fost formată din:

- prof. Daniela Lica, Centrul Județean de Excelență Prahova, Ploiești
- prof. Flavius Boian, Colegiul Național "Spiru Haret" Târgu Jiu
- prof. Florentina Ungureanu, Inspectoratul Școlar Județean Neamț/ Colegiul Național de Informatică Piatra-Neamț
- prof. Marius Nicoli, Colegiul Național "Frații Buzești" Craiova
- prof. Octavian Dumitrașcu, Colegiul Național "Dinicu Golescu" Câmpulung Muscel
- stud. Andrei Arhire - Universitatea Alexandru Ioan Cuza
- stud. Bogdan Iordache - Universitatea din București
- stud. Florian Ușurelu - Universitatea din București/ Universitatea Constantin Brancusi
- stud. Stelian Chichirim - Universitatea din București