

Solutie: problema countfete

Autor:

Stud. Teodor Stelian Ionescu — Universitatea Bucuresti

Solutie $O(2^N)$ ~ 15 puncte

Vom considera **toate subseturile nevide** posibile pentru cele **N** noduri. Pentru fiecare subset in parte vom determina **subarborele conex minim** care include nodurile din respectivul subset. Acest subarbor este caracterizat de toate acele noduri **strict necesare** pentru a putea ajunge intre oricare doua noduri din subsetul considerat si este **unic determinat** de acest subset.

Din subarborele astfel considerat vom alege valoarea minima a unui nod, valoare pe care o vom XOR-a alaturi de valorile tuturor nodurilor considerate in subsetul initial, rezultat pe care il vom aduna la solutie.

Solutii $O(N^2 \cdot \log VALMAX)$, $O(N^2)$ ~ 30-50 puncte

Vom considera pentru simplitate ca toate valorile **v[x]** sunt distincte.

Observatie cheie: Deoarece este vorba de suma XOR, putem trata independent raspunsul pentru fiecare bit in parte al valorilor asociate. De acum inainte vom rezolva problema independent pentru fiecare bit in parte, la final inmultind raspunsul asociat bitului cu indicele **b** cu 2^b si adaugand la raspunsul final.

Idee de rezolvare:

Fixam succesiv fiecare nod din arbore ca fiind nodul cu valoare minima din **subarborele conex minim**. Pentru un nod fixat (notam cu **r**) si un bit **b** ne intereseaza sa calculam pentru cate dintre submultimi subarborii asociati submultimilor respective contin nodul **r**, iar acesta reprezinta intr-adevar minimul din acel subarbor. De asemenea ne intereseaza ca suma XOR a bitului **b** pentru valorile asociate nodurilor considerate in subseturile respective sa fie egala cu NOT valoarea bitului **b** a valorii asociate **minimului** (adica nodului **r**), astfel incat $v[r] \rightarrow b \wedge subset_xor \rightarrow b = 1$. Pentru $v[r] \rightarrow b = 1$, ne vor interesa subseturile cu **val_xor = 0**. Pentru $v[r] \rightarrow b = 0$, ne vor interesa subseturile cu **val_xor = 1**.

Astfel vom considera **arborele inradacinat** in **r** din care vom exclude toti subarborii inradacinati in noduri cu valoare asociata strict mai mare. Acest lucru il putem realiza printr-o simpla parcurgere a arborelui din nodul **r**.

Mai departe observam ca pentru fiecare **fiu** al nodului **r** in **subarborele** astfel obtinut este suficient sa luam in calcul doar **numarul** de noduri ce au valoarea asociata bitului **b** egala cu **0** (notam **A_nod**), respectiv **1** (notam **B_nod**). Mai departe ne vor interesa toate acele submultimi ale acestui subarbor care garanteaza trecerea prin nodul **r** si deci **garantul considerarii minimului**.

Fie functia **nr_mod(valoare_xor, A, B)** = numarul de moduri de a alege **oricate** (cel putin unul) noduri dintr-un subarbor inradacinat astfel incat XOR-ul valorilor asociate sa fie **val_xor** (**0** sau **1** pentru bitul **b**), unde **A** reprezinta numarul de noduri cu **0**, iar **B** numarul de noduri cu **1**.

$$nr_mod(0, A, 0) = 2^A - 1$$

$$nr_mod(0, A, B) = 2^{(A+B-1)} - 1$$

$$nr_mod(1, A, 0) = 0$$

$$nr_mod(1, A, B) = 2^{(A+B-1)}$$

Raspunsul asociat nodului **r** ce reprezinta de fapt numarul subseturilor cu valoarea XOR-ului egala cu **val_xor** dorit si care **garanteaza considerarea minimului** va fi dat de:

nr_mod(val_xor, A_r, B_r) — *SUMA pentru toti fii directi din nr_mod(val_xor, A_fiu, B_fiu)* din principiul includerii si excluderii.

Solutie $O(N * \log N * \log VALMAX)$, $O(N * \log VALMAX) \sim 70-100$ puncte

Urmatoarea imbunatatire se leaga de obtinerea parametrilor **subarborelui inradacinat** nodului **r** la punctul precedent.

Vom considera toate nodurile radacina **r** in **ordine sortata** de la **maxim** la **minim** si vom aplica algoritmul pentru **paduri de multimi disjuncte**, unind mereu in aceeasi multime oricare doua noduri invecinate ce au fost deja parcurse. Fiecare multime va avea asociate valorile **A** si **B** descrise la punctul anterior. Este evident ca la fiecare moment de timp fii directi deja parcursi ai nodului **r** vor fi reprezentantii multimilor fiilor directi ai **subarborelui inradacinat** de la punctul precedent.

Observatie valori distincte:

Observam ca solutia nu se modifica pentru cazul in care exista noduri cu aceeasi valoare asociata **v[x]**. Pur si simplu parcurgem si calculam raspunsurile pentru o ordine arbitrara a nodurilor aflate la egalitate. Acest lucru este corect deoarece putem garanta din implementarea solutiei ca nu vom numara niciodata de doua ori acelasi subset pentru noduri cu valoarea asociata egala. Practic, este ca si cum am asocia valori noi distincte nodurilor cu valori asociate egale.

Solutie $O(N * \log N) \sim 100$ puncte

Solutia finala ce reduce complet factorul de **log VALMAX** din complexitate se bazeaza pe calcularea simultana a raspunsurilor pentru toti **bitii**, lucru ce se poate realiza in mod direct cu ajutorul diverselor operatii pe biti.