

DESCRIEREA SOLUȚIILOR
LOTUL NAȚIONAL DE INFORMATICĂ
CLUJ, 11-18 MAI 2022
BARAJ JUNIORI 1

BÂRLIGĂ!

Propusă de: prof. Emanuela Cerchez, Colegiul Național „Emil Racoviță” Iași

Vom considera că numerele scrise pe scânduri aparțin mulțimii $\{0, 1, \dots, V-1\}$ și că scândurile sunt numerotate de la 0 la $N-1$ (decrementăm numerele la citire).

Numim configurație o modalitate de așezare a scândurilor:

$cf_i = 0$, dacă scândura este cu fața roșie în sus, respectiv 1 în caz contrar.

Putem reține o configurație ca un număr natural, în reprezentarea căruia fiecare bit i ($0 \leq i \leq N-1$) corespunde unei scânduri.

Observăm că pentru fiecare valoare x din mulțimea $\{0, 1, \dots, V-1\}$ există o singură configurație de scânduri pentru care valoarea respectivă nu este vizibilă la finalul jocului (în această configurație setăm bitul i pe valoarea 1 dacă pe scândura i valoarea x este scrisă pe fața roșie, respectiv 0 în caz contrar).

Există 2^N configurații posibile. Vom utiliza un vector de frecvență $nrval$ cu 2^N elemente, unde $nrval_i$ = numărul de valori care nu sunt vizibile pentru configurația i . Determinăm minimul din $nrval$ și configurația $cfmin$ pentru care se obține acest minim. Punctajul maxim va fi $V - nrval_{cfmin}$, iar scândurile care trebuie să fie întoarse sunt cele pentru care bitul corespunzător din $cfmin$ este 1. Mai observăm că există un caz special: dacă $2^N > V$, atunci cu siguranță vor exista configurații pentru care $nrval_{cf} = 0$, deci în acest caz punctajul maxim este V .

Complexitatea acestei soluții este $\mathcal{O}(NV + 2^N)$. Pentru a ne încadra în limita de memorie este necesară o implementare ce folosește operații pe biți.

SECVENȚE

Propusă de: prof. Ionel-Vasile Piț-Rada, Colegiul Național „Traian” Drobeta Turnu Severin

Soluție – Complexitate $\mathcal{O}((N+T)\log N)$.

Citim cele N secvențe speciale și le sortăm lexicografic, complexitatea sortării fiind $\mathcal{O}(N \log N)$. În continuare, o observație **foarte importantă** este de a păstra doar secvențele speciale care NU includ alte secvențe speciale.

Calculăm apoi N sume parțiale astfel:

sp_i = numărul secvențelor super-speciale care au capătul drept în intervalul $[v[i].y; v[i+1].y-1]$ și capătul stâng în intervalul $[1; v[i].x]$.

Apoi pentru fiecare interogare $[L; R]$ cautăm binar secvențele speciale $v[p]$ și $v[q]$ cu proprietățile $L \leq v[p].x$, p minim și respectiv $v[q].y \leq R$, q maxim. Dacă p nu există, atunci răspunsul este 0. În cazul în care există p , atunci calculăm numărul cerut astfel:

$$t = sp[q] - sp[p-1]$$

și avem acum numărate toate secvențele super-speciale cu capăt drept în intervalul $[v[p].y; v[q+1].y-1]$. Din care mai trebuie scăzute cele cu capăt drept din intervalul $[v[q+1].y-1; R+1]$ care conțin secvența $v[q]$:

$$t = t - ((R+1) - v[q+1].y) \cdot v[q].x$$

apoi le eliminăm pe cele care au capătul stâng mai mic decât L :

$$t = t - (R - v[p].y + 1) \cdot (L - 1)$$

SEIF

Propusă de: prof. Ciprian Cheșcă, Liceul Tehnologic „Grigore Moisil” Buzău

Soluția presupune parcurgerea următoarelor etape:

- Determinarea numerelor din fiecare chenar. Al k -lea chenar ($1 \leq k \leq N/2$) are primul număr în colțul stânga-sus (k, k) , apoi numerele până la colțul dreapta-sus $(k, n - k + 1)$, apoi numerele până la colțul dreapta-jos $(n - k + 1, n - k + 1)$, apoi numerele până la colțul stânga-jos $(n - k + 1, k)$ și într-un final numerele până la colțul stânga-sus (k, k) (nu inclusiv).
- Copierea elementelor de pe un cadran într-un vector auxiliar pentru a efectua operația de permutare circulară, în ordinea parcurgerii de mai sus.
- Actualizarea unui vector de rotiri asociat fiecărui chenar în vederea stabilirii numărului final de rotiri al fiecărui chenar. Se va evita în acest mod realizarea unor operații suplimentare. De exemplu dacă avem x rotiri către stânga și apoi x rotiri către dreapta ale aceluiași chenar prin actualizarea vectorilor de rotiri nu se mai efectuează nici o rotire.
- Realizarea în mod eficient a operației de permutare circulară. Dacă vectorul trebuie permutat de x ori într-o direcție, atunci se vor transla direct cu x poziții, ci nu unul câte unul de x ori.
- Copierea elementelor din vectorul permutat auxiliar la loc în matricea – cifru și afișarea matricei în starea finală.

Dacă toate aceste operații au fost executate corect sau au fost implementate operații echivalente cu algoritmul descris mai sus, seiful se va deschide!!!

ECHIPA

Problemele pentru această etapă au fost pregătite de:

- Nicoli Marius – Colegiul Național „Frații Buzești”, Syncro Soft, Craiova
- Boian Flavius – Colegiul Național „Spiru Haret”, Târgu-Jiu
- Bunget Mihai – Colegiul Național „Tudor Vladimirescu”, Târgu-Jiu
- Cerchez Emanuela – Colegiul Național „Emil Racoviță”, Iași
- Cheșcă Ciprian – Liceul Tehnologic „Grigore Mosil”, Buzău
- Frâncu Cristian – Clubul Nerdvana București
- Lica Daniela – Centrul Județean de Excelență Prahova, Ploiești
- Manz Victor – Colegiul Național de Informatică „Tudor Vianu”, București
- Nodea Gheorghe-Eugen – Colegiul Național „Tudor Vladimirescu”, Târgu-Jiu
- Oprea Petru – Liceul „Regina Maria”, Dorohoi
- Pinte Adrian – Inspectoratul Școlar Județean Cluj
- Piț-Rada Ionel-Vasile – Colegiul Național „Traian”, Drobeta Turnu Severin
- Piț-Rada Mihai-Cosmin – Bolt
- Pop Ioan Cristian – Universitatea Politehnica București
- Pracsiu Dan – Liceul Teoretic „Emil Racoviță”, Vaslui
- Șerban Marinel – Colegiul Național „Emil Racoviță”, Iași
- Tulbă-Lecu Theodor-Gabriel – Universitatea Politehnica București
- Voroneanu Radu – Google, Zürich