

Soluție game

Autor prof. Dana Lica

I. Soluție folosind metoda divide et impera

Strategia de rezolvare a problemei pleacă de la problema inițială în care trebuie descoperite ordinea pieselor dintre pozițiile 1 și N . Împărțind problema în două subprobleme identice în care prima întrebare(Q_1) se referă la piesele situate între pozițiile 1 și $\lfloor (N+1)/2 \rfloor$. Dacă $N=7$, atunci primul query se referă la piesele situate pe pozițiile 1,2,3,4.

După prima întrebare se cunosc cele două mulțimi de piese situate în cele două jumătăți $\{1,2,3,4\}$ $\{5,6,7\}$.

Rezolvarea continuă împărțind în jumătate submulțimile obținute anterior, obținând $\{1,2\}$ $\{3,4\}$ respectiv $\{5,6\}$ $\{7\}$ Cu ajutorul unei singure întrebări putem identifica piesele situate în fiecare submulțime nou obținută. Cu alte cuvinte Q_2 ar viza pozițiile 1 2 5 6.

Se continuă după același procedeu până la obținerea unor mulțimi de cardinal 1. $\{1\}$ $\{2\}$ $\{3\}$ $\{4\}$ $\{5\}$ $\{6\}$ $\{7\}$

Ultima întrebare ar fi Q_3 ar fi 1 3 5. Se deduce deci că numărul minim de întrebări este $\lceil \log N \rceil$.

Dacă se păstrează în memorie sub formă de matrice toate întrebările(memorie $N \cdot \log N$) se obține 50 de puncte.

Dacă se ține o listă doar cu capetele stânga dreapta ale submulțimilor curente (memorie $O(N)$) se obțin 70 puncte.

II. Soluția de 100pct

Ideea de rezolvare pleacă de la modalitatea de verificare a corectitudinii unei liste de întrebări. Presupunem că avem o astfel de listă.

Vom verifica dacă în urma răspunsurilor la aceste întrebări se poate deduce exact ce element se află pe poziția i ($1 \leq i \leq n$).

Se pleacă de la presupunerea ca pe poziția i poate sta orice element de la 1 la N , care formează mulțimea curentă de valori candidate(Mc). Parcurgând pe rând întrebările, vom întâlni două cazuri:

- 1) poziția i este vizată în cadrul unei întrebări. Notăm cu Lr mulțimea formată din valorile primite ca răspuns la întrebarea curentă. Atunci mulțimea curentă de candidați pentru poziția i este $Mc = Mc \cap Lr$.
- 2) poziția i nu este vizată în cadrul întrebării curente. Atunci mulțimea curentă de candidați pentru poziția i este $Mc = Mc \setminus Lr$ (diferența)

După parcurgerea tuturor întrebărilor dacă Mc conține un singur element atunci el reprezintă valoarea de pe poziția i . În caz contrar lista de întrebări nu a putut decide aranjarea pieselor.

Pentru toate pozițiile i ($1 \leq i \leq n$) avem nevoie de o matrice $Mask[i][1..logN]$ în care dacă elementul $Mask[i][j]$ este 1 dacă întrebarea cu numărul j se încadrează în cazul 1 sau 0 în caz contrar.

Dacă două linii ale matricei $Mask$ – fie ele i_1 și i_2 sunt identice atunci întrebările puse nu pot determina valorile din pozițiile i_1 și i_2 .

Revenind la soluția problemei, renunțăm la matrice în favoarea unui vector în care valoarea din poziția i are în baza doi reprezentarea corespunzătoare din matricea $Mask$ descrisă anterior. Din cele prezentate mai sus rezultă că în acest vector trebuie să avem valori distincte, deci vom avea reprezentările binare ale numerelor de la 0 la $N-1$.

Ca să afișăm lista întrebărilor, parcurgem fiecare întrebare j de la 1 la $\log N$ și verificăm pentru care poziții i $Mask[i][j]=1$, caz în care se include poziția i în mulțimea pozițiilor întrebării j .

Asfel soluția este $N \log N$, dar memoria poate fi redusă la $O(1)$ folosind operațiile pe biți.

Exemplu $N=5$

$Mask[1]=000_2=0_{10}$

$Mask[2]=001_2=1_{10}$

$Mask[3]=010_2=2_{10}$

$Mask[4]=011_2=3_{10}$

$Mask[5]=100_2=4_{10}$

Întrebarea 1 conține pozițiile 2 și 4

Întrebarea 2 conține pozițiile 3 și 4

Întrebarea 3 conține poziția 5