

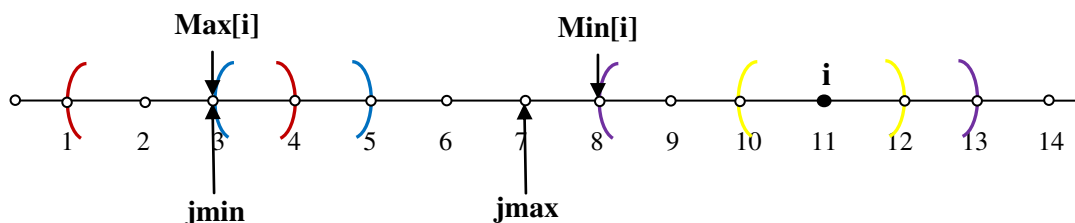
Problema nesiie - Descrierea soluției

Autor: prof. Constantin Gălățan

C. N. „Liviu Rebreanu” Bistrița

Soluție $O(n \cdot n)$ – 50 puncte

În exemplul de mai jos, sunt $T = 4$ turiști, care acoperă intervalele: $[1, 4]$, $[3, 5]$, $[8, 12]$ și $[10, 12]$.



Fie $nr[i]$, numărul maxim de puncte (zile) din intervalul $[1, i]$ în care Nessie poate ieși la suprafață. Presupunem că s-au calculat valorile pentru $nr[j]$ cu j în intervalul $[1, i - 1]$. Observăm că dacă dorim ca Nessie să iasă la suprafața lacului în ziua i , atunci, nu pot fi luate în considerare zilele $j = 8$, $j = 9$ și $j = 10$ deoarece interzic ziua $i = 11$. De asemenea, este suficient să considerăm drept utile doar valori $nr[j]$ doar pentru care $j \geq 3$, unde j este poziția de început al celui mai apropiat interval care nu-l conține pe i (intervalul $[3, 5]$).

Altfel spus, pentru fiecare zi i , trebuie să găsim două valori: $Max[i]$ și $Min[i]$. Prima reprezintă capătul din stânga al celui mai apropiat interval care nu-l conține pe i , iar a doua reprezintă capătul cel mai din stânga al unui interval care-l conține pe i . Se determină apoi $nr[i]$ ca fiind maximul valorilor $nr[j]$ din intervalul $jmin = Max[i]$ și $jmax = Min[i] - 1$, la care se adaugă 1 (pentru ieșirea la suprafață din ziua i). Dacă $jmin > jmax$, atunci în ziua i Nessie nu poate ieși la suprafață.

Dacă pentru fiecare poziție i se parcurg toate valorile j între $jmin$ și $jmax$, atunci complexitatea soluției este $O(n \cdot n)$. Șirurile Max și Min pot fi determinate în $O(n \cdot \log n)$ dacă se sortează intervalele sau în $O(n)$, observând că $Min[i]$ depinde de $Min[i + 1]$ și că $Max[i]$ depinde de $Max[i - 1]$.

Soluție $O(n \cdot \log n)$ – 70 puncte

Valorile din șirul nr pot fi determinate în $O(n)$ observând că pentru două zile i_1 și i_2 dacă $i_1 < i_2$, atunci $jmin[i_1] \leq jmin[i_2]$ și $jmax[i_1] \leq jmax[i_2]$. Prin urmare intervalele $[imin, imax]$ asociate poziției i se deplasează uniform spre dreapta, odată cu i . Aceasta ne sugerează ideea utilizării unui *deque* în care menținem poziția maximului valorilor $nr[j]$ din intervalul asociat, $[imin, imax]$. Cum fiecare poziție j intră și iese din coadă o singură dată, complexitatea determinării șirului nr scade la $O(n)$.

Dacă șirurile Min și Max se determină în urma sortării intervalelor, atunci complexitatea finală a algoritmului este dată de algoritmul de sortare: $O(n \cdot \log n)$.

Soluție $O(n)$ – 100 puncte

Soluția în complexitate $O(n)$ depinde de observațiile făcute anterior, combinând determinarea în complexitate $O(n)$ a șirurilor Min și Max , cu determinarea în complexitate $O(n)$ a șirului nr .