

Tabăra de pregătire a Lotului Național de Informatică

Câmpulung Muscel, 3-10 Mai 2018

Baraj 4 – Seniori

Sursa: anagrame.cpp



Problema anagrame

100 puncte

Se consideră două numere naturale nenule N și K ($K \leq N$). Un șir de N numere naturale conține numere de la 1 la K . Fiecare valoare de la 1 la K apare în șir cel puțin o dată. Se cunosc elementele șirului, dar nu și ordinea acestora.

Cerință

Se cere să reconstituiți șirul folosind anumite informații despre acesta.

Va trebui să implementați o funcție

void Solve(int NrTest, int N, int *S)

Graderul comisiei va apela funcția o singură dată și va transmite prin intermediul parametrilor următoarele informații (vezi tabelul de restricții de mai jos):

NrTest = numărul testului

N = numărul de elemente din șir

S = un șir conținând cele N elemente ale șirului căutat în ordine crescătoare.

În funcția **Solve()** veți apela următoarele funcții puse la dispoziție de graderul comisiei.

int Ask(int *X)

transmite prin parametrul X un șir conținând elementele șirului căutat într-o ordine aleasă de voi și returnează numărul minim de **swap**-uri între elemente de pe poziții consecutive care pot fi aplicate pe șirul X pentru a-l transforma în șirul cerut.

Această funcție poate fi apelată de un număr limitat de ori (conform tabelului de restricții).

int GiveSolution(int *X)

transmite prin parametrul X șirul căutat.

Această funcție va fi apelată o singură dată la final, după toate apelurile funcției **Ask()**.

Exemplu

Acțiune	Explicație
Solve(0,4,{1,2,3,4})	Șirul care trebuie reconstituit este {3,2,1,4}
Ask({1,2,3,4})	Returnează 3 <ul style="list-style-type: none">- swap (2,3) – {1,3,2,4}- swap (1,3) – {3,1,2,4}- swap (1,2) – {3,2,1,4}
Ask({2,4,3,1})	Returnează 3 <ul style="list-style-type: none">- swap (4,3) – {2,3,4,1}- swap (2,3) – {3,2,4,1}- swap (4,1) – {3,2,1,4}
GiveSolution({3,2,1,4})	Șirul reconstituit este {3,2,1,4}

Tabăra de pregătire a Lotului Național de Informatică

Câmpulung Muscel, 3-10 Mai 2018

Baraj 4 – Seniori

Sursa: `anagrame.cpp`



Structura testelor

NrTest	N	Nr. max. apeluri	Observații	Punctaj
0	2^2	24	exemplu	0
1	2^4	2^8	șirul e o permutare	7
2	2^5	2^{10}		
3	2^5	$2^5 * 5 * 2$	șirul e o permutare	6
4	2^6	$2^6 * 6 * 2$		
5	2^6	$2^6 * 6$	șirul e o permutare	14
6	2^7	$2^7 * 7$		
7	2^9	$2^9 * 2$	șirul conține doar elementele 1 și 2	11
8	2^{10}	$2^{10} * 2$		
9	2^9	$2^9 * 2$	șirul e o permutare	19
10	2^{10}	$2^{10} * 2$		
11	2^9	$2^9 * 2$		18
12	2^{10}	$2^{10} * 2$		
13	2^9	2^9	șirul e o permutare	11
14	2^{10}	2^{10}		
15	2^9	2^9		14
16	2^{10}	2^{10}		

Observații referitoare la verdictele de evaluare

Dacă numărul de apeluri depășește valoarea maximă alocată pe test, puteți primi unul dintre următoarele verdicte:

- 400 (max 100) apeluri ale funcției Ask
- > 500 (max 100) apeluri ale funcției Ask

Verdictele de mai sus nu garantează corectitudinea răspunsului găsit

Limită de timp: 0,5 secunde

Memorie totală disponibilă: 512 MB