

Suceava, 30 aprilie 2019 – 04 mai 2019

Sursa: scara.c, scara.cpp, scara.pas

**Problema 2 - scara**

**100 de puncte**

Avem  $N$  persoane notate cu etichetele  $1, 2, \dots, N$ , într-o ordine oarecare și o scară cu  $N$  trepte.

Persoanele sunt așezate în șir indian, cu fața spre locul unde se află scara. Treptele scării sunt inițial neocupate.

În mod repetat persoana aflată la acel moment la capătul din dreapta al șirului se poziționează pe scară pe prima treaptă neocupată, iar fiecare dintre persoanele aflate pe treptele inferioare coboară de pe scară și se repositionează la capătul din dreapta al șirului, începând cu cea de la prima treaptă a scării. Acțiunea se oprește atunci când sunt ocupate toate treptele pe scară.

Exemplu: Inițial la etapa 0 avem  $\text{sir} = (4, 2, 1, 3)$  și  $\text{scara} = (0, 0, 0, 0)$

Etapa 1:  $(4, 2, 1) / (3, 0, 0, 0)$       Etapa 2:  $(4, 2, 3) / (0, 1, 0, 0)$       Etapa 3:  $(4, 2) / 3, 1, 0, 0)$   
 Etapa 4:  $(4, 3, 1) / (0, 0, 2, 0)$       Etapa 5:  $(4, 3) / 1, 0, 2, 0)$       Etapa 6:  $(4, 1) / (0, 3, 2, 0)$   
 Etapa 7:  $(4) / (1, 3, 2, 0)$       Etapa 8:  $(1, 3, 2) / 0, 0, 0, 4)$       ...  
 Final:  $() / (3, 2, 1, 4)$

**Cerințe**

Se cere să se afișeze etichetele persoanelor în ordinea în care acestea sunt așezate pe scară la final.

**Date de intrare**

În fișierul de intrare **scara.in** se află scris pe prima linie numărul  $N$ . Pe a doua linie se află scrise  $N$  numere naturale separate prin câte un spațiu, reprezentând etichetele celor  $N$  persoane, în ordinea inițială din șir, de la stânga spre dreapta.

**Date de ieșire**

În fișierul de ieșire **scara.out** vor fi scrise pe prima linie și separate prin câte un spațiu cele  $N$  numere naturale reprezentând etichetele persoanelor corespunzătoare așezării pe scară.

**Restricții și precizări**

- $1 \leq N \leq 2000$
- Pentru 22 puncte  $N \leq 15$

**Exemple**

| scara.in     | scara.out | Explicații   |
|--------------|-----------|--|
| 4<br>4 2 1 3 | 3 2 1 4   | $(4, 2, 1, 3) / (0, 0, 0, 0)$ , $(4, 2, 1) / (3, 0, 0, 0)$ ,<br>$(4, 2, 3) / (0, 1, 0, 0)$ , $(4, 2) / 3, 1, 0, 0)$ , $(4, 3, 1) / (0, 0, 2, 0)$ ,<br>$(4, 3) / 1, 0, 2, 0)$ , $(4, 1) / (0, 3, 2, 0)$ , $(4) / (1, 3, 2, 0)$ ,<br>$(1, 3, 2) / (0, 0, 0, 4)$ , $(1, 3) / (2, 0, 0, 4)$ , $(1, 2) / (0, 3, 0, 4)$ ,<br>$(1) / (2, 3, 0, 4)$ , $(2, 3) / (0, 0, 1, 4)$ , $(2) / (3, 0, 1, 4)$ ,<br>$(3) / (0, 2, 1, 4)$ , $() / (3, 2, 1, 4)$ |
| 3<br>2 3 1   | 3 1 2     | $(2, 3, 1) / (0, 0, 0)$ , $(2, 3) / (1, 0, 0)$ , $(2, 1) / (0, 3, 0)$ ,<br>$(2) / (1, 3, 0)$ , $(1, 3) / (0, 0, 2)$ , $(1) / (3, 0, 2)$ , $(3) / (0, 1, 2)$ ,<br>$() / (3, 1, 2)$  |

**Timp maxim de execuție/test: 0.5 secunde – Windows / 0.2 secunde - Linux**

**Memorie totală: 128 MB**

**Dimensiunea maximă a sursei: 20 KB**