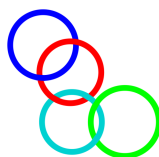


# Inelele parașutei

O versiune timpurie și destul de sofisticată de ceea ce noi acum numim parașută este descrisă în lucrarea lui Leonardo Codex atlanticus (cca. 1485). Parașuta lui Leonardo constă dintr-o pânză cerată, ținută deschisă cu ajutorul unei structuri din lemn în formă de piramidă.

## Inelele legate

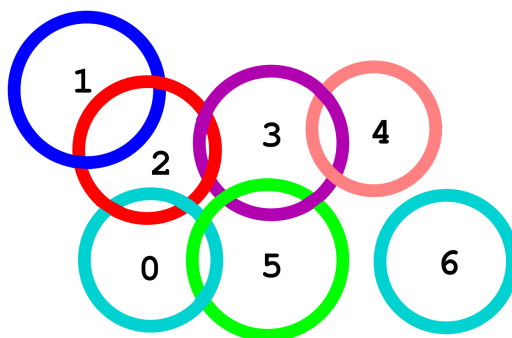
Cu 500 de ani mai târziu, parașutistul Adrian Nicolae a testat proiectul lui Leonardo. În acest scop, o structură ușoară modernă leagă parașuta lui Leonardo de corpul uman. Această structură este formată din carabine în formă de inele, confecționate dintr-un material rezistent. Inelele pot fi ușor legate între ele, iar fiecare inel poate fi închis sau deschis din nou. Numim lanț o secvență de unul sau mai multe inele legate, în care fiecare inel este legat cu alte două inele, cu excepția primului și ultimului, care sunt legate cu câte un singur inel, după cum este ilustrat mai jos. Prin definiție, un singur inel este, de asemenea, un lanț.



Evident, întrucât un inel poate fi legat nu numai cu două, dar cu trei și chiar mai multe inele, sunt posibile și alte configurații. Spunem că un inel este critic, dacă după deschiderea și eliminarea acestuia, inele rămase formează un set de lanțuri disjuncte sau alte inele nu mai există. Prin alte cuvinte, lipsa de inele este și ea un lanț.

## Exemplu

Se consideră cele 7 inele din figura ce urmează, numerotate de la 0 la 6. Există două inele critice. Unul din inelele critice este cel cu numărul 2. După eliminarea acestui inel, inelele rămase formează lanțurile [1], [0, 5, 3, 4] și [6]. Un alt inel critic este cel cu numărul 3. După eliminarea acestuia, inelele rămase formează lanțurile [1, 2, 0, 5], [4] și [6]. Dacă am elimina un oricare alt inel, nu vom obține un set de lanțuri disjuncte. De exemplu, deși după eliminarea inelului 5 vom obține lanțul [6], inelele legate 0, 1, 2, 3 și 4 nu formează un lanț.



## Sarcină

Trebuie să elaborezi un program care calculează numărul de inele critice din configurația dată.

La început, există un anumit număr de inele disjuncte. După aceea, inele sunt legate împreună. La un moment dat, se cere să returnezi numărul de inele critice din configurația curentă. Mai concret, trebui să implementezi trei rutine.

- `Init(N)` — la început aceasta se apelează exact o singură dată pentru a afla numărul  $N$  de inele disjuncte din configurația inițială, numerotate de la 0 la  $N - 1$  (inclusiv).
- `Link(A, B)` — cele două inele  $A$  și  $B$  vor fi legate împreună. Este garantat că  $A$  și  $B$  sunt diferite și nu sunt încă legate în mod direct. Nu există alte restricții suplimentare față de  $A$  și  $B$ , în particular, restricții ce ar rezulta din constrangeri fizice. Evident, apelurile `Link(A, B)` și `Link(B, A)` sunt echivalente.
- `CountCritical()` — returnează numărul de inele critice din configurația curentă.

## Exemplu

Se consideră  $N = 7$  inele, care, inițial, sunt deconectate. Vom arata o secvență posibilă de apelurile, după ultimul din care se va obține configurația prezentată în figura de mai sus.

Apelul	Valorile returnate
<code>Init(7)</code>	
<code>CountCritical()</code>	7
<code>Link(1, 2)</code>	
<code>CountCritical()</code>	7
<code>Link(0, 5)</code>	
<code>CountCritical()</code>	7
<code>Link(2, 0)</code>	
<code>CountCritical()</code>	7
<code>Link(3, 2)</code>	
<code>CountCritical()</code>	4
<code>Link(3, 5)</code>	
<code>CountCritical()</code>	3
<code>Link(4, 3)</code>	
<code>CountCritical()</code>	2

## Subtask 1 [20 de puncte]

- $N \leq 5\,000$ .
- Funcția `CountCritical` este apelată doar o singură dată, după toate celelate apeluri; funcția `Link` este apelată de cel mult 5 000 de ori.

## Subtask 2 [17 puncte]

- $N \leq 1\,000\,000$ .
- Funcția `CountCritical` este apelată doar o singură dată, după toate celelate apeluri; funcția `Link` este apelată de cel mult 1 000 000 ori.

## Subtask 3 [18 puncte]

- $N \leq 20\,000$ .
- Funcția `CountCritical` este apelată de cel mult de 100 de ori; funcția `Link` este apelată de cel mult 10 000 ori.

## Subtask 4 [14 puncte]

- $N \leq 100\,000$ .
- Funcțiile `CountCritical` și `Link` sunt apelate în total de cel mult 100 000 de ori.

## Subtask 5 [31 de puncte]

- $N \leq 1\,000\,000$ .
- Funcțiile `CountCritical` și `Link` sunt apelate în total de cel mult 1 000 000 de ori.

## Detalii de implementare

Tu trebuie să transmiți exact un singur fișier, denumit `rings.c`, `rings.cpp` sau `rings.pas`. Acest fișier implementează subprogramele descrise mai sus utilizând signaturile ce urmează.

### Programele în limbajul C/C++

```
void Init(int N);
void Link(int A, int B);
int CountCritical();
```

## Programele în limbajul Pascal

```
procedure Init(N : LongInt);  
procedure Link(A, B : LongInt);  
function CountCritical() : LongInt;
```

Aceste subprograme trebuie să se comporte așa cum este descris mai sus. Desigur, ești liber să implementezi pentru uzul intern al acestora și alte subprograme. Submit-urile tale nu trebuie să interacționeze în nici într-un fel cu intrarea / ieșirea standard, și nici cu oricare alt fișier.

### Model de evaluator

Modelul de evaluator citește intrarea în formatul ce urmează:

- linia 1: N, L;
- liniile 2, ..., L + 1:
  - -1 apelează `CountCritical`;
  - A, B parametrii pentru `Link`.

Modelul de evaluator va tipări toate rezultatele returnate de `CountCritical`.