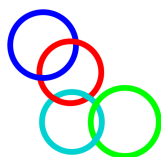


Langevarjurõngad

Leonardo da Vinci kirjeldas oma teoses *Codex Atlanticus* (ca 1485) varajast, kuid juba üsna hästi välja arendatud versiooni seadmest, mida me tänapäeval tunneme langevarju nime all. Leonardo langevari koosnes linasest riidest, mida hoidis avatuna püramiidikujuline puitraam.

Seotud rõngad

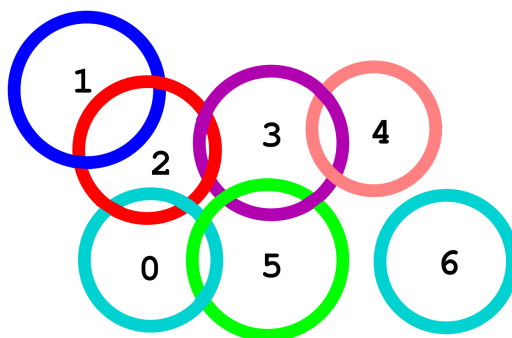
Enam kui 500 aastat hiljem testis Leonardo projekti langevarjur Adrian Nicholas. Selleks kinnitati Leonardo langevari tema külge moodsa kerge struktuuriga. Selles struktuuris kasutame seotud rõngaid, mis on ühtlasi ka kanga kinnitusaasadeks. Rõngad on tehtud tugevast, kuid painduvast materjalist ja neid kerge omavahel siduda, kuna nad on avatavad ja taassuletavad. Rõngaste teatud konfiguratsioone nimetame *ahelateks*. *Ahel* on rõngaste jada, milles iga rõngas on ühendatud (ülimalt) kahe naabriga, nagu näha alloleval joonisel. Sellisel ahelal peab kindlasti olema algus ja lõpp (rõngad, mis on ühendatud ainult ühe naabriga). Erijuhuna loeme ahelaks ka üksiku rõnga.



Ilmselt on võimalikud ka muud konfiguratsioonid, kuna üks rõngas võib olla seotud ka kolme või enama teise rõngaga. Ütleme, et rõngas on *kriitiline*, kui selle avamisel ja konfiguratsioonist eemaldamisel moodustavad kõik ülejäänud rõngad mingi hulga ahelaid (või kui rohkem rõngaid ei olegi). Teisisõnu, pärast kriitilise rõnga eemaldamist ei tohi alles jääda midagi, mis ei ole ahel.

Näide

Vaatleme alloleval joonisel kujutatud 7 rõngast, mis on nummerdatud 0 kuni 6. Selles konfiguratsioonis on kaks kriitilist rõngast. Üks neist on rõngas number 2: selle eemaldamise järel moodustavad ülejäänud rõngad ahelad [1], [0, 5, 3, 4] ja [6]. Teine kriitiline rõngas on 3: selle eemaldamisel moodustavad ülejäänud rõngad ahelad [1, 2, 0, 5], [4] ja [6]. Ülejäänud rõngastest ükskõik millise eemaldamisel me sellist lõikumatumest ahelatest koosnevat hulka ei saa. Näiteks rõnga number 5 eemaldamise järel on [6] küll ahel, aga omavahel seotud rõngad 0, 1, 2, 3 ja 4 ahelat ei moodusta.



Ülesanne

Kirjutada programm, mis loendab kriitiliste rõngaste arvud talle kirjeldatud konfiguratsioonides.

Algne konfiguratsioon koosneb hulgast omavahel sidumata rõngastest. Seejärel hakatakse rõngaid ühendama. Igal hetkel võidakse küsida kriitiliste rõngaste arvu jooksvas konfiguratsioonis. Täpsemalt tuleb realiseerida kolm alamprogrammi.

- `Init(N)` — kutsutakse välja üks kord programmi täitmise alguses; teatab, et algses konfiguratsioonis on N omavahel sidumata rõngast, mis on nummerdatud 0 kuni $N-1$ (kaasa arvatud).
- `Link(A, B)` — rõngad number A ja B ühendatakse omavahel; on teada, et A ja B on erinevad ning ei ole varem vahetult ühendatud; rohkem piiranguid A ja B kohta ei ole (muuhulgas ei arvestata mingeid füüsi(ka)lisi kitsendusi); `Link(A, B)` ja `Link(B, A)` on ilmselt samaväärsed.
- `CountCritical()` — tagasta jooksva konfiguratsiooni kriitiliste rõngaste arv.

Näide

Oletame, et meie joonisel ($N = 7$) on rõngad alguses omavahel ühendamata. Vaatleme järgnevat tegevuste jada, mille lõpuks on rõngad joonisel kujutatud konfiguratsioonis.

Kutse	Tagastab
<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

Alamülesanne 1 [20 punkti]

- $N \leq 5\,000$.
- Funktsiooni `CountCritical` kutsutakse välja ainult üks kord, pärast kõiki teisi kutseid; alamprogrammi `Link` kutsutakse välja ülimalt 5 000 korda.

Alamülesanne 2 [17 punkti]

- $N \leq 1\,000\,000$.
- Funktsiooni `CountCritical` kutsutakse välja ainult üks kord, pärast kõiki teisi kutseid; alamprogrammi `Link` kutsutakse välja ülimalt 1 000 000 korda.

Alamülesanne 3 [18 punkti]

- $N \leq 20\,000$.
- Funktsiooni `CountCritical` kutsutakse välja ülimalt 100 korda; alamprogrammi `Link` kutsutakse välja ülimalt 10 000 korda.

Alamülesanne 4 [14 punkti]

- $N \leq 100\,000$.
- Alamprogramme `CountCritical` ja `Link` kutsutakse kokku välja ülimalt 100 000 korda.

Alamülesanne 5 [31 punkti]

- $N \leq 1\,000\,000$.
- Alamprogramme `CountCritical` ja `Link` kutsutakse kokku välja ülimalt 1 000 000 korda

Realisatsioon

Lahendusena tuleb esitada üks fail nimega `rings.c`, `rings.cpp` või `rings.pas`. See fail peab realiseerima eelpool kirjeldatud alamprogrammid vastavalt alltoodud signatuuridele.

C/C++ programmid

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

Pascali programmid

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

Need alamprogrammid peavad töötama nagu eelpool kirjeldatud. Muidugi võib nende realiseerimiseks kirjutada ka teisi sisemisi alamprogramme. Lahendus ei tohi pöörduda standardsisendi, standardväljundi ega ühegi teise faili poole.

Lokaalne hindaja

Lokaalne hindaja ootab sisendit järgmises vormingus:

- 1. rida: N, L;
- read 2, ..., L+1:
 - -1 — kutsuda välja funktsioon `CountCritical`;
 - A, B — alamprogrammi `Link` parameetrid.

Lokaalne hindaja väljastab funktsiooni `CountCritical` tagastatud väärtused.