



## Problema 2 - echilateral

100 puncte

O rețea triunghiulară de latură  $n$  se obține descompunând un triunghi echilateral de latură  $n$  în triunghiuri echilaterale de latură 1, folosind drepte paralele la laturile triunghiului inițial. De exemplu în figurile de mai jos avem rețele triunghiulare de latură 4. Numim *noduri ale rețelei* vârfurile triunghiurilor de latură 1 folosite în descompunere. Astfel pe prima rețea am desenat un triunghi echilateral cu vârfuri în nodurile rețelei iar pe a doua rețea am desenat două triunghiuri echilaterale cu vârfuri în noduri.



### Cerință

Să se scrie un program care pentru  $n$ ,  $a$ , și  $b$  cunoscute, determină numărul de triunghiuri echilaterale cu vârfurile în nodurile unei rețele de latură  $n$  care au lungimile laturilor cuprinse între valorile  $a$  și  $b$ .

### Date de intrare

Fișierul de intrare `echilateral.in` conține pe prima linie numerele  $n$ ,  $a$  și  $b$  separate prin câte un spațiu.

### Date de ieșire

Fișierul de ieșire `echilateral.out` va conține pe prima linie numărul de triunghiuri echilaterale cu vârfurile în nodurile unei rețele de latură  $n$  care au lungimile laturilor cuprinse între valorile  $a$  și  $b$ , modulo 666013.

### Restricții și precizări

- $1 \leq n \leq 1.000.000.000$ ;
- $1 \leq a \leq b \leq 1.000.000$ ;

### Exemplu

echilateral.in	echilateral.out	Explicație
4 1 2	29	<p>Avem 16 triunghiuri de latură 1 (cele care acoperă rețeaua).</p> <p>Mai avem încă 6 triunghiuri de latură <math>\sqrt{3}</math> (similar triunghiului din a doua figură și având una dintre laturi verticală).</p> <p>Mai avem încă 7 triunghiuri de latură 2.</p> <p>În total avem <math>16+6+7=29</math> triunghiuri.</p>

**Timp maxim de execuție/test: 0.4 secunde**

**Total memorie disponibilă: 64 MB**

**Dimensiunea maximă a sursei: 10KB**