

Coduri

Fișiere sursă: `coduri.pas` sau `coduri.c` sau `coduri.cpp`

Un detectiv particular are de rezolvat un caz special. Este vorba de o deturnare de fonduri. Pentru a putea rezolva cazul trebuie să găsească un șir cu n coduri distincte. Fiecare cod este un număr natural scris în baza 10. Din păcate lucrurile nu sunt simple, pentru că din cercetările efectuate a obținut două informații. Prima informație este legată de faptul că suma pătratelor codurilor este un cub perfect, iar a doua spune că suma cuburilor codurilor este un pătrat perfect.

Cerință

Ajutați detectivul să găsească un șir de coduri x_1, x_2, \dots, x_n , care verifică condițiile din enunț și $x_i \leq n^{14}$, pentru orice i cu $1 \leq i \leq n$.

Date de intrare

Fișierul de intrare `coduri.in` conține pe prima linie numărul natural n .

Date de ieșire

Fișierul de ieșire `coduri.out` va conține n linii, câte una pentru fiecare cod din șir, în ordine crescătoare.

Restricții

- $1 \leq n \leq 20$

Exemplu

<code>coduri.in</code>	<code>coduri.out</code>
2	625 1250

Timp maxim de execuție: 1 secundă/test

Logic

100 puncte

Demonstrarea automată a teoremelor și verificarea satisfiabilității unei formule constituie două capitole importante în cadrul logicii matematice. Formulele propoziționale sunt alcătuite din variabile propoziționale (variabile care pot lua doar două valori: sau adevărat sau fals) și din operatorii logici și, sau, negație, echivalent, implică.

Iată câteva exemple de formule propoziționale:

$$\sim p \& (q \Leftrightarrow p) \Rightarrow q$$

$$p \mid q \Leftrightarrow \sim p \& \sim q$$

$$p$$

$$p \Rightarrow q \Rightarrow a \Rightarrow t \Rightarrow \sim p$$

În acest exemplu, p și q sunt variabilele propoziționale, \sim este operatorul unar negație, $\&$ este operatorul binar și, \mid este operatorul binar sau, \Rightarrow este implicația logică (și apare numai în acest sens, nu și \Leftarrow), iar \Leftrightarrow este echivalența logică. În plus, într-o formulă propozițională pot să apară și paranteze care stabilesc ordinea operațiilor. În lipsa parantezelor, operatorii în ordinea priorității lor sunt \sim & \mid \Rightarrow \Leftrightarrow .

În formulele de forma „ $A_1 \text{op} A_2 \text{op} \dots \text{op} A_K$ ” asociațiile se fac de la dreapta la stânga (adică „ $A_1 \text{op} (A_2 \text{op} (\dots \text{op} A_K) \dots)$ ”), unde **op** este unul dintre $\&$, \mid , \Rightarrow sau \Leftrightarrow și A_i sunt formule propoziționale, cu i de la 1 la K .

În general, o formulă propozițională se definește astfel:

- orice variabilă propozițională este formulă propozițională
- dacă A și B sunt formule propoziționale, atunci (A) , $\sim A$, $A \& B$, $A \mid B$, $A \Rightarrow B$, $A \Leftrightarrow B$ sunt formule propoziționale.

Dacă înlocuim într-o formulă propozițională toate variabilele cu valori de adevăr (adevărat sau fals), obținem o afirmație. Valoarea de adevăr a unei afirmații este dată de următoarea definiție:

- dacă afirmația constă dintr-o singură valoare de adevăr, afirmația ia valoarea respectivă
- dacă A și B sunt afirmații, atunci:

A este adevărată dacă și numai dacă valoarea sa de adevăr este adevărat

(A) este adevărată dacă și numai dacă A este adevărată

$\sim A$ este falsă dacă și numai dacă A este adevărată

$A \& B$ este adevărată dacă și numai dacă atât A cât și B sunt adevărate

$A \mid B$ este falsă dacă și numai dacă A este fals și B este fals

$A \Rightarrow B$ este adevărată dacă și numai dacă $\sim A \mid B$ este adevărată

$A \Leftrightarrow B$ este adevărată dacă și numai dacă $(A \Rightarrow B) \& (B \Rightarrow A)$ este adevărată.

Se numește soluție a formulei propoziționale P (formulă în care apar numai variabilele propoziționale distincte A_1, A_2, \dots, A_N) orice N -uplu (v_1, v_2, \dots, v_N) (cu v_i valori de adevăr) pentru care înlocuind fiecare variabilă A_i cu valoarea v_i , afirmația rezultată este adevărată.

Cerință

Logica fiind un obiect nesuferit de studenții de la informatică, ei apelează la informaticienii din clasa a IX-a pentru a-i ajuta să numere câte soluții distincte are o formulă propozițională dată.

Date de intrare

În fișierul de intrare **logic.in** se găsește o formulă propozițională unde variabilele propoziționale sunt reprezentate de litere mici ale alfabetului englez.

Date de ieșire

În fișierul de ieșire **logic.out** se va afișa numărul de soluții pentru formula propozițională din fișierul de intrare, urmat de caracterul sfârșit de linie.

Restricții:

- La intrare se va da întotdeauna o formulă propozițională corectă sintactic
- Formula are mai puțin de 232 de caractere
- În formulă nu apar mai mult de 10 litere mici ale alfabetului latin

Exemple:

logic.in	logic.out
$p \mid q \Leftrightarrow \sim p \& \sim q$	4
a	1

Timpul de rulare/test: maxim 1 secundă.

Recomandare. Afișarea soluției se face în modul următor:

Pascal: `writeln(f, nsol);`

C: `fprintf(f, "%d\n", nsol);`

C++: `f << nsol << "\n";`

Poligon

100 puncte

Fișiere sursă: poligon.pas sau poligon.c sau poligon.cpp

Se dă un caroiaj de $M \times N$ în care sunt plasate K puncte. Fiecare punct poate fi legat de vecinul sau direct pe maxim opt direcții (N, NE, E, SE, S, SV, V, NV).

Cerință

Determinați patrulateralele având vârfurile în punctele date iar laturile formate din legături între două sau mai multe puncte coliniare.

Date de intrare

Fișier de intrare: **poligon.in**

Linia 1: $M \ N \ K$ //trei numere naturale nenule, separate prin câte un spațiu, reprezentând dimensiunile M, N ale

//caroiajului și K numărul de puncte.

Linia 1: $I_1 \ J_1 \ V_1$ //trei numere naturale nenule, separate printr-un spațiu, reprezentând coordonatele //punctului 1 și respectiv direcțiile spre care este legat de vecini direcți.

Linia 2: $I_2 \ J_2 \ V_2$ //trei numere naturale nenule, separate printr-un spațiu, reprezentând coordonatele

//punctului 2 și respectiv direcțiile spre care are vecini direcți.

...

Linia K: $I_k \ J_k \ V_k$ //trei numere naturale nenule, separate printr-un spațiu, reprezentând coordonatele

//punctului K și respectiv direcțiile spre care are vecini direcți.

Codificarea direcțiilor se face printr-un număr cuprins între 0 și 255. Reprezentarea binară a acestuia pe 8 cifre reprezintă, începând de la stânga spre dreapta, legătura pe direcțiile: (1- legatura, 0 – nu)

N NE E SE S SV V NV

De exemplu:

1 0 0 0 0 1 1 0 = 134 deci legături spre N, SV, V

Date de ieșire

Fișier de ieșire: **poligon.out**

Linia 1: n_{pol} //număr natural reprezentând numărul patrulaterelor.

Restricții

- $1 < M, N \leq 100$
- $4 \leq K \leq 50$

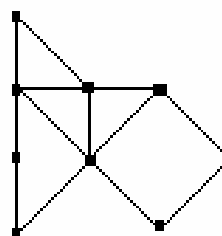
Exemplu

POLIGON.IN

```
4 4 9
1 1 24
2 1 184
2 2 43
2 3 22
3 1 136
3 2 213
3 4 5
4 1 192
```

POLIGON.OUT

```
6
```



Olimpiada Națională de Informatică
Clasa a IX-a
Ziua 1



4 3 65

Timp maxim de executare/ test: 2 secunde