

Clasa a IX-a

# coduri 100 puncte

## Fișiere sursă: coduri.cpp, coduri.c, coduri.pas

Întorcându-se de la școală în ziua în care a aflat cum se face înmulțirea numerelor, Gigel a auzit la televizor următoarea afirmație: "Pentru a face avere, nu trebuie să aduni bani în viață, ci trebuie să-i înmulțești".

Toate acestea l-au pus pe gânduri, așa că s-a hotărât să inventeze propriul "sistem de codificare" pentru numere reale mai mari decât 0 care să aibă următoarele proprietăți:

- fiecare număr va fi codificat sub forma unui șir de valori întregi (pozitive și/ sau negative)
- dacă un număr real x are codul  $c_x$  și un număr real y are codul  $c_y$ , atunci numărul real rezultat prin înmulțirea lui x și y trebuie să aibă codul obținut prin "adunarea" codurilor  $c_x$  și  $c_y$ .
- dacă un număr real x se poate scrie ca produs de numere  $y_1, y_2, ..., y_k$ , atunci codul lui x se obține prin "adunarea" codurilor numerelor  $y_1, y_2, ..., y_k$ .

Considerăm un cod  $c_1$  format din  $n_1$  valori  $a_{n_1}$  ..  $a_1$  și un cod  $c_2$  format din  $n_2$  valori  $b_{n_2}$  ... $b_1$ , atunci codul  $c_3$  obținut prin "adunarea" codurilor  $c_1$  și  $c_2$  va avea  $n_3$  valori  $d_{n_3}$  ..  $d_1$ , cu proprietățile următoare:

- n<sub>3</sub> este maximul dintre n<sub>1</sub> şi n<sub>2</sub>

$$d_{i} = \begin{cases} a_{i} + b_{i} , \text{ pentru } i = 1, ..., \text{minim}(n_{1}, n_{2}) \\ a_{i} , \text{ pentru } i = \text{minim}(n_{1}, n_{2}) + 1, ..., n_{1} \text{ dacă minim}(n_{1}, n_{2}) = n_{2} \\ b_{i} , \text{ pentru } i = \text{minim}(n_{1}, n_{2}) + 1, ..., n_{2} \text{ dacă minim}(n_{1}, n_{2}) = n_{1} \end{cases}$$

#### Cerință

Dându-se N numere reale mai mari strict decât 0, să se scrie codificarea acestora în sistemul inventat de Gigel.

#### Date de intrare

Fişierul de ieşire **coduri**.in va conține:

- pe prima linie din fișier se află numărul **N** de numere reale
- pe următoarele **N** linii cele **N** numere reale, fiecare pe câte o linie.

### Date de ieșire

Fișierul de ieșire coduri.out va conține N linii:

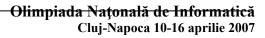
pe linia i (i între 1 și N): numărul de valori folosite pentru codificarea numărului cu indicele i din fișierul de intrare, urmat de un spațiu și apoi valorile ce alcătuiesc codul numărului, separate două câte două printr-un singur spațiu.

## Restricții și precizări

- 2 ≤ N ≤ 18
- Separatorul între partea întreagă și partea zecimală este virgula.
- Orice număr are după virgulă cel mult 5 cifre.
- Valorile din codurile numerelor din fisierele de test trebuie să fie cuprinse în intervalul [-106, 106].
- Partea întreagă a fiecărui număr real este o valoare mai mică sau egală cu 20000.
- Toate numerele din fisierele de test sunt strict pozitive și distincte două câte două.
- Numărul maxim de valori utilizat pentru codificarea unui număr este 2500.
- Dacă există mai multe soluții de codificare, se va afișa una singură.
- Nu trebuie să existe două numere diferite cu aceeași codificare.
- 40% din teste vor conține numai numere întregi, 30% din teste vor conține numere întregi și numere reale fără perioadă și 30% din teste vor conține numere întregi și numere reale cu și fără perioadă.

## Exemplu

coduri.in	coduri.out	Explicație	
-----------	------------	------------	--





## Clasa a IX-a

8 10 2 5 0,3 7 2,1 1,(7) 1,2(34)	2 1 1 3 -1 0 1 3 1 1 0 3 2 1 0 3 -1 2 1 3 1 3 1 2 1 11 2 1 2	10=2*5, iar suma codurilor pentru 2 și 5, determină codul lui 10 2,1=7*0,3, iar suma codurilor pentru 7 și 0,3 determină codul lui 2,1
--	---	--

Timp maxim de execuţie/test (Windows/Linux): 0.2 secunde