

Test recapitulativ clasa a IX-a

SUBIECTUL I – Algoritmi elementari

Două numere se numesc **oglundite** dacă fiecare se obține din celălalt, prin parcurgerea cifrelor acestuia de la dreapta la stânga. Două numere se numesc **impar-oglundite** dacă numerele obținute din acestea, prin îndepărtarea tuturor cifrelor lor pare, sunt oglindite.

Date de intrare: de la tastatură se citește n ($1 \leq n \leq 1000$) și apoi de pe fiecare din următoarele n linii se citesc două numere naturale x și y ($1 \leq n \leq 10^9$).

Date de ieșire: se afișează un număr natural reprezentând numărul de perechi de numere x și y care sunt impar oglundite.

Exemplu:

Intrare	ieșire	Explicație
5 <u>523</u> 84 <u>356</u> 102 31 <u>101</u> <u>211</u> 523 84536 <u>968</u> <u>209</u>	3	Perechile scrise cu rosu formeaza perechi impar-oglundite

SUBIECTUL II – Tablouri unidimensionale, tablouri bidimensionale

Scrieți un program C/C++ care citește de la tastatură numere naturale: n ($n \in [2, 20]$), k ($k \in [2, n]$) și $n \cdot n$ numere din intervalul $[0, 10^9]$, elemente ale unui tablou bidimensional cu n linii și n coloane. Programul transformă tabloul în memorie, interschimbând șirul elementelor situate pe linia a k -a, în stânga diagonalei principale, parcurse de la stânga la dreapta, cu șirul elementelor situate pe coloana a k -a, deasupra diagonalei principale, parcurse de sus în jos, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, fiecare linie pe câte o linie a ecranului, cu elementele fiecărei linii separate prin câte un spațiu

Exemplu:

Intrare	ieșire	Explicație
5 4 2 4 3 5 6 8 0 9 8 7 2 6 9 0 5 6 1 3 6 9 7 3 9 4 2	2 4 3 6 6 8 0 9 1 7 2 6 9 3 5 5 8 0 6 9 7 3 9 4 2	

SUBIECTUL III – probleme de optim

Se consideră șirul 1, 3, 7, 13, 21, 31, 43 ... definit astfel: $f_0=1$, iar $f_n=f_{n-1}+2 \cdot n$, dacă $n \geq 1$ (unde n este un număr natural).

Se citesc de la tastatură două numere naturale din intervalul $[1, 10^9]$, x și y ($x < y$), reprezentând doi termeni aflați pe poziții consecutive în șirul dat, și se cere să se scrie în fișierul text **bac.out**, separați prin câte un spațiu, toți termenii șirului mai mici sau egali cu y , în ordine inversă a apariției lor în șir.

Proiectați un **algorithm eficient** din punctul de vedere al spațiului de memorie și al timpului de executare.

Exemplu:

intrare	bac.txt	Explicație
21 31	31 21 13 7 3 1	