

Algoritmi elementari cu vectori

1. Obiectivele lucrării

În această lucrare se studiază vectorii (tablouri unidimensionale), modul de instanțiere, modul de transmitere al acestor vectori ca parametrii într-o funcție și se dau ca exemple atât vectori de numere de tip *int* cât și alte tipuri de date.

2. Breviar teoretic

O structură de date este un aranjament al datelor în memorie. Un program scris într-un limbaj de programare este alcătuit din una sau mai multe tipuri de structuri de date și algoritmi de procesare a datelor prezente în structurile folosite. Cea mai simplă structură de date o reprezintă **vectorii**.

Vectorii sunt tablouri de date unidimensionale. Numele vectorului este o referință către zona de memorie unde se stochează componentele vectorului. Componentele vectorului se accesează prin indexare.

Un vector se instanțiază conform sintaxei:

tipDate numeVector[dimensiune];

unde *dimensiune* = o constantă (număr natural);

Exemplu de instanțiere:

`int a[100]; char a[10].`

Componentele vectorului sunt: $a[0], a[1], a[2], \dots, a[9], \dots, a[n-1]$.

Un vector poate fi creat și prin listarea directă a valorilor lui inițiale:

Exemplu: `int a[3] = { 10, 20, 30 };`

3. Probleme propuse

Problema 1

Citim un număr natural N . Citim N numere întregi într-un vector a . Să calculăm și să afișăm maximul și minimul din vector.

Problema 2

Citim un număr natural N . Citim N numere întregi într-un vector a . Să calculăm și să afișăm suma elementelor vectorului a .

Problema 3

Citim un număr natural N . Citim N numere întregi într-un vector a . Să calculăm și să afișăm dacă toate elementele sunt egale între ele.

Problema 4

Citim un număr natural N . Citim N numere întregi într-un vector a . Să calculăm și să afișăm câte elemente pare și câte elemente impare sunt în vector.

Problema 5

Citim un număr natural N . Citim N numere întregi într-un vector a . Să calculăm și să afișăm dacă un număr x , citit de la tastatură, este prezent în vector.

Problema 6

Citim un număr natural N . Citim N numere naturale într-un vector a . Să calculăm și să afișăm care cifră apare de cele mai multe ori în scrierea celor N numere.

Exemplu: Dacă $N = 3$ și numerele sunt: 77, 4234, 3337 atunci cifra 3 apare de cele mai multe ori (de 4 ori.)

Indicație: Se va folosi un vector auxiliar `nrAparitiiCifra[]`, ce are 10 componente. Astfel, în `nrAparitiiCifra[0]` se va memora numărul total de apariții al cifrei 0, etc.

Problema 7

Citim un număr natural N . Citim N numere naturale într-un vector a . Să se sorteze în ordine crescătoare elementele vectorului.

Problema 8

Citim un număr natural N . Citim N numere naturale într-un vector a . Să se verifice dacă elementele vectorului constituie un palindrom.

Exemplu: 12321 sau 1221 -> Palindrom

Problema 9

Să se elaboreze un algoritm pentru ștergerea dintr-un vector a elementului cu indicele k și deplasarea elementelor vectorului, începând cu indicele $k+1$, cu o poziție spre stânga, lungimea logică a vectorului micșorându-se cu 1.

Problema 10

Să se elaboreze un algoritm pentru inserarea într-un vector a unei valori x pe poziția k cu deplasarea elementelor vectorului spre dreapta.

MATRICE

Problema 11

Să se elaboreze un algoritm pentru adunarea a două matrice de aceeași dimensiune (maxim 3). Exemplu:

Prima matrice:	1 2
	3 4
A doua matrice:	5 6
	7 8
Rezultatul adunării:	6 8
	10 12

Problema 12

Să se elaboreze un algoritm pentru adunarea elementelor de pe diagonala principala a unei matrice de dimensiunea 3×3 .

Problema 13

Să se elaboreze un algoritm pentru calcularea determinantului unei matrice de ordin 3.

Problema 14

Se citește de la tastatură o matrice de numere întregi, de dimensiuni cunoscute (numărul de linii și coloane). Se citesc în variabilele $lin1$ și $lin2$, numerele a două linii din matrice. Să se comute în matrice aceste două linii între ele.