

**1. Subprogramul `schimba_jum` are doi parametri**

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul modifică ordinea elementelor astfel încât, numerele din jumătatea a doua a vectorului să fie schimbate cu cele din prima jumătate. Dacă vectorul are un număr impar de elemente, atunci poziția elementului din mijlocul tabloului rămâne neschimbată.

*Exemplu:* dacă  $n=7$  și  $v=(1,45,20,26,13,50,12)$ , atunci, după reordonare,  $v=(13,50,12,26,1,45,20)$ .

**2. Subprogramul `inserare_div` are doi parametri**

- **v**, un tablou unidimensional ce conține numere naturale, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul inserează după fiecare număr par nenul, cel mai mare divizor diferit de numărul însuși și actualizează valoarea lui n.

*Exemplu:* dacă  $n=5$  și  $v=(13,50,12,7,20)$ , atunci, după inserare,  $v=(13,50,25,12,6,7,20,10)$  și  $n=8$ .

**3. Subprogramul `prod_max` are doi parametri**

- **v**, un tablou unidimensional ce conține numere întregi, cu cel mult 4 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul returnează cel mai mare produs care se poate obține înmulțind două numere din vector.

*Exemplu:* dacă  $n=5$  și  $v=(-3,5,-20,7,2)$ , atunci subprogramul returnează valoarea 60.

**4. Subprogramul `min_fmax` are doi parametri**

- **v**, un tablou unidimensional ce conține numere naturale, cu cel mult 4 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul returnează cel mai mic număr din vector dintre cele care apar de cele mai multe ori.

*Exemplu:* dacă  $n=7$  și  $v=(3,5,20,7,5,2,20)$ , atunci subprogramul returnează valoarea 5.

**5. Subprogramul `ordoneaza` are doi parametri**

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul ordonează crescător prima jumătate a tabloului și descrescător cea de-a doua jumătate a vectorului. Dacă vectorul are un număr impar de elemente, atunci poziția elementului din mijlocul tabloului rămâne neschimbată.

*Exemplu:* dacă  $n=7$  și  $v=(10,40,2,26,13,50,12)$ , atunci, după reordonare,  $v=(2,10,40,26,50,13,12)$ .

**6. Subprogramul `secv_max` are patru parametri**

- **v**, un tablou unidimensional ce conține numere naturale, cu cel mult 4 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v;
- **k**, prin care furnizează lungimea celei mai lungi secvențe de numere egale;
- **x**, prin care furnizează cea mai mică valoare care se află într-o secvență de lungime k, de numere egale.

*Exemplu:* dacă  $n=13$  și  $v=(3,5,5,5,4,4,8,8,8,9,2,2,2)$ , atunci  $k=3$  și  $x=2$ .

**7. Subprogramul `cresc_max` are doi parametri:**

- **v**, un tablou unidimensional ce conține numere naturale, cu cel mult 4 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din v.

Subprogramul returnează lungimea celei mai lungi secvențe de numere aflate în ordine strict crescătoare.

*Exemplu:* dacă  $n=9$  și  $v=(3,5,4,7,9,16,6,8,9)$ , atunci subprogramul returnează 4.

**8.** Subprogramul **fibonacci** are doi parametri:

- **n**, prin care primește o valoare naturală  $2 < n < 100$ ;
- **v**, un tablou unidimensional prin care furnizează șirul format cu primele  $n$  numere din șirul lui Fibonacci.

*Exemplu:* dacă  $n=5$ , subprogramul furnizează  $v=(1, 1, 2, 3, 5)$ .

**9.** Subprogramul **prime** are doi parametri:

- **n**, prin care primește o valoare naturală  $2 < n < 100$ ;
- **v**, un tablou unidimensional prin care furnizează șirul format cu primele  $n$  numere prime.

*Exemplu:* dacă  $n=5$ , subprogramul furnizează  $v=(2, 3, 5, 7, 11)$ .

**10.** Subprogramul **grupe** are trei parametri:

- **n**, prin care primește o valoare naturală  $2 < n < 10$ ;
- **v**, prin care furnizează un tablou unidimensional ce conține șirul crescător în care fiecare număr  $i \in [1, n]$  apare de exact  $i$  ori;
- **m**, prin care furnizează numărul de elemente din tabloul  $v$ .

*Exemplu:* dacă  $n=4$ , subprogramul furnizează  $v=(1, 2, 2, 3, 3, 3, 4, 4, 4, 4)$  și  $m=10$ .

**11.** Subprogramul **sum** are doi parametri:

- **v**, un tablou unidimensional ce conține numere naturale, cu cel mult 4 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ , reprezentând numărul de elemente din  $v$ .

Subprogramul returnează valoarea primului element din vector cu proprietatea că este egal cu suma tuturor elementelor aflate pe poziții anterioare lui. Dacă în vector nu există un astfel de număr, subprogramul returnează -1.

*Exemplu:* dacă  $n=6$  și  $v=(3, 5, 8, 2, 18, 9)$ , atunci subprogramul returnează 8.

**12.** Subprogramul **divizorN** are doi parametri

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ .

Subprogramul returnează cel mai mare divizor comun al celor  $n$  numere din tabloul  $v$ .

*Exemplu:* dacă  $n=6$  și  $v=(10, 450, 255, 650, 500, 120)$ , atunci valoarea returnată este 5.

**13.** Subprogramul **progresie** are doi parametri

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ .

Subprogramul verifică dacă elementele vectorului pot fi rearanjate astfel încât acestea să formeze o progresie aritmetică cu rație pozitivă. În caz afirmativ returnează rația progresiei iar altfel returnează 0.

*Exemplu:* dacă  $n=6$  și  $v=(10, 13, 7, 4, 19, 16)$ , atunci valoarea returnată este 3.

**14.** Subprogramul **multime** are doi parametri

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ .

Subprogramul verifică dacă elementele vectorului formează o mulțime (toate elementele sunt distincte). În caz afirmativ returnează valoarea 1 iar altfel returnează 0.

*Exemplu:* dacă  $n=6$  și  $v=(10, 13, 7, 4, 19, 16)$  atunci valoarea returnată este 1, iar dacă  $v=(10, 13, 7, 13, 19, 16)$ , atunci valoarea returnată este 0.

**15.** Subprogramul **inserare** are doi parametri

- **v**, un tablou unidimensional ce conține numere naturale nenule, cu cel mult 9 cifre;
- **n**, prin care primește o valoare naturală  $2 < n < 100$ .

Subprogramul inserează după fiecare element din vector un nou element, egal cu cifra sa maximă.

*Exemplu:* dacă  $n=3$  și  $v=(120, 4, 1239)$ , atunci, valorile furnizate sunt  $n=6$  și  $v=(120, 2, 4, 4, 1239, 9)$ .