

Laborator 3 – Programare Procedurala (Mate)

Săptămâna a 3-a

1. **(1p)** Se citește un număr natural n de la tastatură.
 - a) Să se determine cel mai mare număr ce poate fi format cu cifrele lui n .
 - b) Să se determine numerele de 3 cifre mai mici decât n^2 care au suma divizorilor un număr de tip palindrom (12321 este un palindrom).

2. **(4p)** Se citește o valoare n de la tastatură și apoi se citesc n numere naturale.
 - a) Să se calculeze suma elementelor pare de pe poziții impare.
 - b) Să se determine elementul minim și elementul maxim și să se interschimbe.
 - c) Să se afișeze pozițiile pe care se afla o valoare k citită de la tastatură.
 - d) Să se determine câte numere au cifra zecilor o valoare din intervalul (2,8).
 - e) Să se determine câte numere au proprietatea de a fi palindrom.
 - f) Să se afișeze numerele care au cel mult 6 divizori.
 - g) Să se calculeze media aritmetică a numerelor pare și să se insereze între prima și a doua valoare a șirului.
 - h) Să se calculeze media aritmetică a numerelor și să se insereze între ultima și penultima valoare a șirului.

3. **(2p)** Fie un șir de n ($0 < n < 1000$) numere întregi. Se cere:
 - a) Să se verifice dacă numerele din șirul dat pot forma o mulțime (elementele sunt distincte).
 - b) Să se scrie în ordine crescătoare factorii primi din descompunerea elementului maxim din șirul dat.

4. **(3p)** Se dau două mulțimi prin doi vectori (toate elementele vectorilor sunt distincte).
 - a) Să se verifice dacă un întreg aparține celor două mulțimi.
 - b) Determinați intersecția, reuniunea și diferența a celor două mulțimi.

5. **(10p)** Se consideră o secvență de numere naturale x_1, x_2, \dots, x_n . Din această secvență se pot obține alte secvențe folosind următoarea operație: se extrage elementul de pe poziția i ($i > 1$), se mută toate elementele situate la stânga poziției i cu o poziție la dreapta, iar elementul de pe poziția i se plasează pe prima poziție a secvenței.
 - (a) Să se scrie un program care primind o secvență de numere naturale x_1, x_2, \dots, x_n afișează toate secvențele care se pot obține din aceasta folosind o singură dată operația definită mai sus. Ordinea în care sunt afișate secvențele rezultate nu contează. De exemplu, din secvența 1, 2, 3 folosind o singură operație, mutând elementul de pe poziția 2 se obține secvența 2, 1, 3 și mutând de pe poziția 3 se obține secvența 3, 1, 2.

(b) Să se scrie un program care primind două permutări x_1, x_2, \dots, x_n și y_1, y_2, \dots, y_n ale multimii $\{1, \dots, n\}$ afișează secvență minimă de operații de tipul de mai sus prin care permutarea x_1, x_2, \dots, x_n se poate transforma în permutarea y_1, y_2, \dots, y_n . O operație va fi afișată prin acel element x_i care se mută pe prima poziție. De exemplu dacă se primesc permutările: 4, 5, 6, 7, 8, 9, 3, 1, 2 și 4, 9, 6, 5, 7, 8, 3, 1, 2 secvența optimă este: 6, 9, 4 adică din prima permutare se extrage 6 și se pune în față, apoi se extrage 9 și se pune în față, iar apoi se extrage 4 și se pune în față.