

Tutoriat 4 - Programarea Algoritmilor

1. Scrieți o funcție care verifică dacă două șiruri de caractere sunt anagrame sau nu

utilizând un singur vector de frecvențe.

2. Scrieți o funcție care returnează numărul de modificări necesare pentru a face 2 șiruri de caractere anagrame.

Exemplu: abcc, abcd -> 1

3. Fișierul text *exemplu.txt* conține un text pe mai multe linii, cuvintele fiind despărțite între ele prin spații și semnele de punctuație uzuale. Implementați un program care să scrie în fișierul text *grupe_cuvinte.txt* cuvintele distincte (fără duplicate) din fișierul dat, grupate după lungimile lor. Grupele se vor scrie în fișier în ordinea descrescătoare a lungimilor cuvintelor, iar în fiecare grupă cuvintele vor fi scrise în ordine alfabetică.

4. Fișierul text *numere.txt* conține numere naturale pe mai multe linii, numerele fiind despărțite între ele prin spații. Scrieți un program care să afișeze cel mai mare și cel mai mic număr care se poate obține din toate cifrele tuturor numerelor din fișier. Rezolvați problema în cel puțin 3 moduri diferite, respectiv folosind șiruri de caractere, liste, mulțimi și dicționare!

5. Scrieți o funcție recursivă care calculează factorialul unui număr dat:

Exemplu: $\text{factorial}(3) = 6$

6. Scrieti o functie care calculeaza al n-lea numar fibonacci, folosind functii recursive.

Exemplu: fibonacci(4) = 3

7. Folosind functiile recursive, scrieti o functie care returneaza elementul maxim dintr-o lista.

8. Implementati o functie care converteste un string in int, folosind recursia.

9. Implementati o functie care sa gaseasca prima cifra dintr-un numar dat, folosind recursia.

10. Implementati o functie recursiva care verifica daca un element exista intr-o lista sortata, folosind cautarea binara.