

Laborator 10

• Programarea calculatoarelor –

1. Să se definească o structură cu următoarele date despre un student:

- id - număr întreg;
- nume - un șir de maxim 20 caractere;
- prenume - un șir de maxim 20 caractere;
- nota - un număr real.

Scrieți câte o funcție pentru fiecare cerință:

a. să se citească datele de la tastatură și să se memoreze într-un fișier binar (transmis ca parametru; citirea se încheie cu EOF (Ctrl+Z in Windows, Ctrl+D in Linux). **void**

creare(char *nume_fisier_binar)

b. să se citească datele din fișierul binar și să se afișeze într-un alt fișier

text: **(serializare/deserializare)**

void afisare (char *nume_fisier_binar, char *nume_fisier_text)

c. pentru un id dat, să se modifice nota studentului în fișierul binar:

void modificare (char *nume_fisier_binar)

d. să se adauge date pentru un nou student în fișierul binar:

void adaugare (char *nume_fisier_binar)

2. Definiți o structură Angajat care să permită memorarea următoarelor informații despre un angajat: **CNP, nume, vârstă și salariu**. Numele se va păstra sub forma unui șir de caractere alocat dinamic. Scrieți câte o funcție pentru rezolvarea fiecăreia dintre următoarele cerințe:

a) scrierea în fișierul binar angajati.bin a informațiilor despre mai mulți angajați cu datele citite de la tastatură;

b) afișarea pe ecran a datelor despre un angajat din fișierul binar angajati.bin pentru care se cunoaște codul numeric personal;

c) adăugarea unui nou angajat în fișierul binar angajati.bin;

d) scrierea în fișierul text salarii.txt a numelui și salariului fiecărui angajat din fișierul binar angajati.bin, precum și salariul mediu;

e) afișarea pe ecran a numelor angajaților din fișierul binar angajati.bin care au salariul egal cu cel maxim ;

f) ștergerea unui angajat din fișierul binar angajati.bin pentru care se cunoaște codul numeric personal.

3. Fișierul text *siruri.txt* are următoarea structură: pe prima linie un număr natural nenul n_1 , pe a doua linie n_1 numere întregi despărțite între ele prin câte un spațiu, pe a treia linie un număr natural nenul n_2 , pe a patra linie n_2 numere întregi despărțite între ele prin câte un spațiu ș.a.m.d.

a) Folosind funcțiile pentru alocarea/realocarea dinamică a memoriei, creați o structură de date bidimensională care să permită memorarea tuturor șirurilor de numere întregi din fișierul text dat. Pe fiecare linie, prima valoare va reprezenta numărul de valori de pe linia respectivă, iar următoarele valori vor fi chiar valorile din șirul respectiv.

b) Scrieți o funcție care să afișeze o structură de date de tipul anterior.

c) Folosind funcția `qsort` din biblioteca `stdlib.h`, sortați crescător elementele fiecărui șir, iar apoi sortați șirurile în ordinea crescătoare a lungimilor lor.

d) Scrieți o funcție care să interclaseze două șiruri de numere întregi alocate dinamic și sortate crescător. Rezultatul interclasării se va reține în primul șir, iar cel de-al doilea șir va fi șters din memorie!

e) Folosind funcția definită anterior, interclasați toate șirurile de numere întregi date astfel încât rezultatul să fie memorat pe prima linie a structurii de date bidimensionale create la punctul a)

f) Adăugați la sfârșitul fișierului text *siruri.txt*, pe o linie nouă, elementele șirului obținut prin interclasarea șirurilor inițiale și apoi eliberați toată memoria alocată dinamic.