Ministerul Educației și Cercetării al Republicii

Moldova Universitatea Tehnică a Moldovei

Colegiul Universității Tehnice a Moldovei

RAPORT

Lucrarea de laborator nr. 1

la „Programarea Calculatoarelor”

Tema: Prelucrarea tipurilor de date lista

Elaborat de:

Bogatu Alexei, gr. DAW-241

Verificat de:

prof., dl Mihail Potlog

Chișinău, 2025

**Tema: Prelucrarea tipurilor de date lista**

***1. Scopul lucrării:***

1. Însuşirea procedeelor de creare a unei structuri dinamice de date de tip listă, conform specificațiilor propuse;
2. Însuşirea operațiilor specifice tipurilor dinamice de date:
3. Afișarea, căutarea, permutarea, eliminarea, adăugarea datelor în cadrul unei structuri
4. dinamice de date de tip listă;
5. Elaborarea algoritmilot pentru tipuri de date de tip listă;
6. Translarea și implimentarea algoritmilor pentru tipuri de date de tip listă în limbajul de programare.

***2. Condiţia lucrării:***

Lucrarea 3: Să se creeze o listă ce conține datele despre titlul, autorul precum și editura unor cărți. Să se afișeze lista la ecran după care să se afișeze acele cărți din listă care sunt editate în *București*.

***3. Textul programului în limbajul C++ :***

#include <iostream>

#include <string>

using namespace std;

struct carte {

string titlu;

string autor;

string editura;

};

struct nod {

carte a;

nod \*urm;

};

int main() {

nod \*prim, \*ultim, \*nou, \*p;

int n;

cout << "Cate carti doriti sa mai adaugati: ";

cin >> n;

if(n==0) {

cout << "Nu sunt carti.";

return 0;

}

prim = new nod;

cout << "Introduceti prima carte:" << endl;

cout << "1) Titlu: "; cin.ignore(); getline(cin, prim->a.titlu);

cout << " Autor: "; getline(cin, prim->a.autor);

cout << " Editura: "; getline(cin, prim->a.editura);

prim->urm = NULL;

ultim = prim;

ultim->urm = NULL;

if(n>1) {

for(int i=2; i<=n; i++) {

nou = new nod;

cout << i << ") Titlu: "; getline(cin, nou->a.titlu);

cout << " Autor: "; getline(cin, nou->a.autor);

cout << " Editura: "; getline(cin, nou->a.editura);

nou->urm = NULL;

ultim->urm = nou;

ultim = nou;

}

}

cout<<endl;

int o = 1;

p = prim;

cout << "Lista de carti:" << endl;

while(p) {

cout << o << ") Titlu: " << p->a.titlu << endl;

cout << " Autor: " << p->a.autor << endl;

cout << " Editura: " << p->a.editura << endl;

o++;

p=p->urm;

}

cout<<endl;

o = 1;

p = prim;

cout << "Lista de carti de la editura Bucuresti:" << endl;

while(p) {

if(p->a.editura == "Bucuresti") {

cout << o << ") Titlu: " << p->a.titlu << endl;

cout << " Autor: " << p->a.autor << endl;

cout << " Editura: " << p->a.editura << endl;

o++;

}

p=p->urm;

}

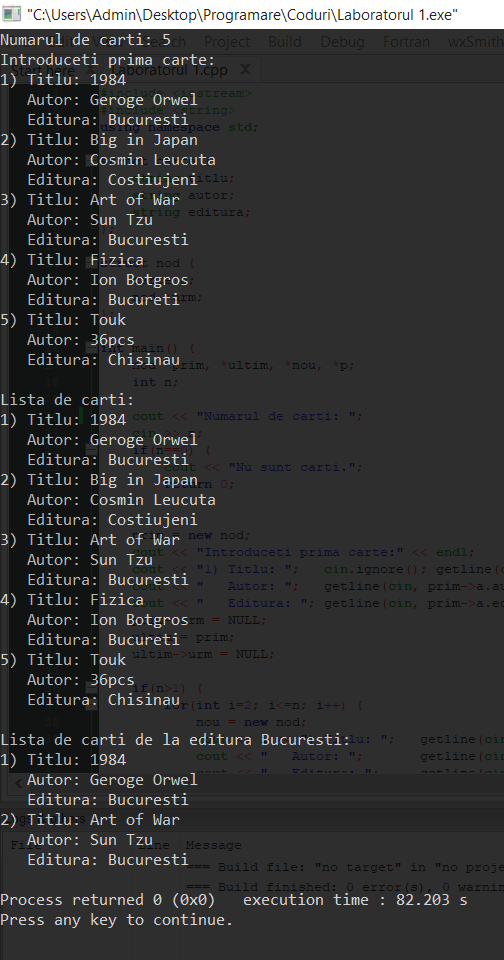
delete prim, ultim, nou, p;

return 0;

}

***4. Schema bloc a programului:***

***5. Rezultatele obţinute în urma execuţiei programului:***



***6. Concluziile referitor la îndeplinirea lucrării.***

1. Am însuşit procedeele de creare a unei structuri dinamice de date de tip listă, conform specificațiilor propuse;
2. Am însuşit operațiile specifice tipurilor dinamice de date care sunt: afișarea, căutarea, permutarea, eliminarea, adăugarea datelor în cadrul unei structuri dinamice de date de tip listă;
3. Am elaborat un algorit pentru tipul de date de tip listă;
4. Am translat și implimentat algoritmii pentru tipuri de date de tip listă în limbajul de programare C++.