**Ministerul Educației al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică și Microelectronică**

**Departamentul Ingineria Software și Automatică.**

**Raport**

Lucrarea de laborator nr.1

Disciplina: Programarea Orientata pe Obiect

Tema: Structura – mecanism de abstractizare.

Varianta: 5.

**Efectuat**: st.gr. TI-207 Bunescu Gabriel

**Verificat**: Lisnic Inga

Chișinău 2021

**Scopul lucrării:**

**•** Studierea programării prin abstractizarea datelor;

• Studierea regulilor de definire şi utilizare a structurilor de date;

• Crearea variabilelor de tip structură, accesarea componentelor unei structuri.

**Sarcina**

**Varianta:5**

Să se creeze tipul abstract de date (structura) – computer, care se compune din informaţia despre firma producătoare, procesor, frecvenţa de tact, etc. să se definească funcţiile de setare a datelor, de modificare a informaţiei, de comparaţie. Pentru iniţializarea componentelor textuale să se utilizeze operatorul de alocare dinamică a memoriei new. Să se elibereze memoria alocată. În funcţia main să se exemplifice sortarea tabloului de tip calculator după cîmpurile frecvenţa de tact + procesor.

**Codul C++:**

#include <iostream>

#include <string.h>

using namespace std;

typedef struct products {

char name[50];

int quantity;

float price;

int cpu\_gen;

int cpu\_frecv;

}products;

products \*psoft;

int n;

void aloc\_memory() {

cout << "\n-----------------------------------" ;

cout << "\n>> Introdu numarul de calculatoare:";

cin >> n;

if(n > 1) {

psoft = new products[n];

} else

cout << ">> Nu ai introdus corect numarul de elemente <<";

}

void scan\_product() {

for(int i = 0; i < n; i++) {

cout << "\nVa rugam sa introduceti numele computerului :";

cin.getline(psoft[i].name,50);

cin.getline(psoft[i].name,50);

cout << "\nAcum, va rugam sa introduceti cantitatea de calculatoare:";

cin >> psoft[i].quantity;

cout << "\nAcum, va rugam sa introduceti pretul computerelor:";

cin >> psoft[i].price;

cout << "\nIntroduceti generarea CPU (de la 1 la 10):";

cin >> psoft[i].cpu\_gen;

cout << "\nIntroduceti frecventa procesorului:";

cin >> psoft[i].cpu\_frecv;

}

}

void print\_elements () {

for (int i = 0; i < n; i++ ) {

cout << "\n>> |Denumirea " << i + 1 << " :" << psoft[i].name ;

cout << "\n>> |Cantitatea " << i + 1 << " :" << psoft[i].quantity ;

cout << "\n>> |Pretul " << i + 1 << " :" << psoft[i].price ;

cout << "\n>> |CPU " << i + 1 << " :" << psoft[i].cpu\_gen ;

cout << "\n>> |Frecventa " << i + 1 << " :" << psoft[i].cpu\_frecv ;

}

}

void sort\_price() {

for(int i = 0; i < n; i++) {

for(int j = i + 1; j < n; j++)

if(psoft[i].price > psoft[j].price) {

products temp = psoft[i];

psoft[i] = psoft[j];

psoft[j] = temp;

}

}

}

void modif\_element() {

char name[50];

cout << "\n>> Introdu numele computerului:" ;

cin.getline(name,50);

cin.getline(name,50);

for (int i = 0; i < n; i ++) {

if(strcmp(name,psoft[i].name) == 0){

cout << ">> Introdu numele:";

cin.getline(psoft[i].name, 50);

cout << ">>Introdu Cantitatea:";

cin >> psoft[i].quantity;

cout << "\n>> Introdu pretul:";

cin >> psoft[i].price;

cout << "\n>> Introdu generarea CPU:";

cin >> psoft[i].cpu\_gen;

cout << "\n>> Introdu Frecventa:";

cin >> psoft[i].cpu\_frecv;

}

}

}

void search\_element\_name() {

char name[50];

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

cout << " \n>>Introdu denumirea calculatorului care il cauti:";

cin.getline(name,50);

cin.getline(name,50);

for(int i = 0; i < n; i++) {

if((strcmp(name,psoft[i].name) == 0)) {

cout << "\n>> Denumirea " << i + 1 << " :" << psoft[i].name;

cout << "\n>> Cantitatea " << i + 1 << " :" << psoft[i].quantity;

cout << "\n>> Pretul " << i + 1 << " :" << psoft[i].price;

cout << "\n>> CPU " << i + 1 << " :" << psoft[i].cpu\_gen;

cout << "\n>> Frecventa " << i + 1 << " :" << psoft[i].cpu\_frecv;

}

}

}

void search\_element\_price() {

int price\_p;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

cout << "\n>> Introdu pretul:";

cin >> price\_p;

for(int i = 0; i < n; i++) {

if(price\_p == psoft[i].price) {

cout << ">> |Denumirea" << i + 1 << ":" << psoft[i].name;

cout << ">> |Cautarea" << i + 1 << ":" << psoft[i].quantity;

cout << ">> |Pretul" << i + 1 << ":" << psoft[i].price;

cout << ">> |CPU" << i + 1 << ":" << psoft[i].cpu\_gen;

cout << ">> |Frecventa" << i + 1 << ":" << psoft[i].cpu\_frecv;

}

}

}

void menu() {

cout << "-------------------------------------------------------" << endl;

cout << ">>Va rugam sa alegeti una dintre urmatoarele operatiuni" << endl;

cout << "-------------------------------------------------------" << endl;

cout << "|1| |Alocam memorie |" << endl;

cout << "|2| |Introducera datelor |" << endl;

cout << "|3| |Afisarea datelor |" << endl;

cout << "|4| |Sortare dupa pret |" << endl;

cout << "|5| |Cautarea unui element dupa nume | " << endl;

cout << "|6| |Cautarea unui element dupa pret |" << endl;

cout << "|7| |Modificarea unui element | " << endl;

cout << "|8| |Eliberarea momorie | " << endl;

cout << "|0| |EXIT | " << endl;

cout << "-------------------------------------------------------" << endl;

}

void delet\_memory() {

delete[] psoft;

psoft = nullptr;

}

int main(int argc, const char \* argv[]) {

int a;

menu();

while(1) {

cout << "\n-----------------" ;

cout << "\nIntrodu un numar:" ;

cin >> a;

switch (a) {

case 1:

cout << "\nMemoria a fost alocata" ;

aloc\_memory();

break;

case 2:

cout << ">> Introducerea datelor" ;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

scan\_product();

break;

case 3:

cout << "\n>> Afisarea calculatoarelor:" ;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

print\_elements();

break;

case 4:

cout << "Sortarea dupa pret a fost efectuata cu success";

sort\_price();

break;

case 5:

cout << "\nCautarea unui element dupa nume" ;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

search\_element\_name();

break;

case 6:

cout << "\nCautarea unui element dupa pret" ;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

search\_element\_price();

break;

case 7:

cout << "\nModificarea datelor unui calculator" ;

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

modif\_element();

break;

case 8:

cout << "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

cout << "\nMemoria a fost eliberata cu success" ;

delet\_memory();

break;

case 0:

cout << "Ati esit cu success -\_-";

exit(0);

break;

default:

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" ;

cout << "Eroare nu ai introdus numarul corect.";

exit(0);

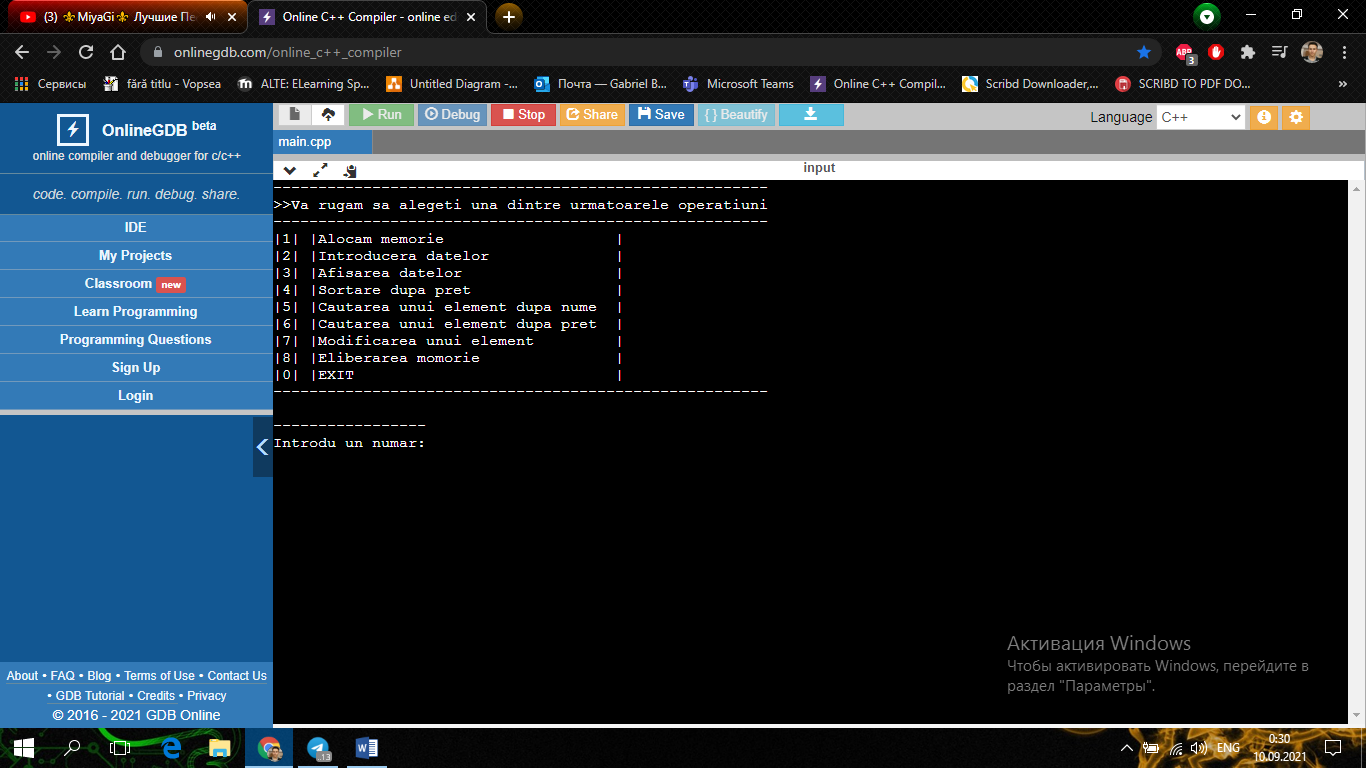
}

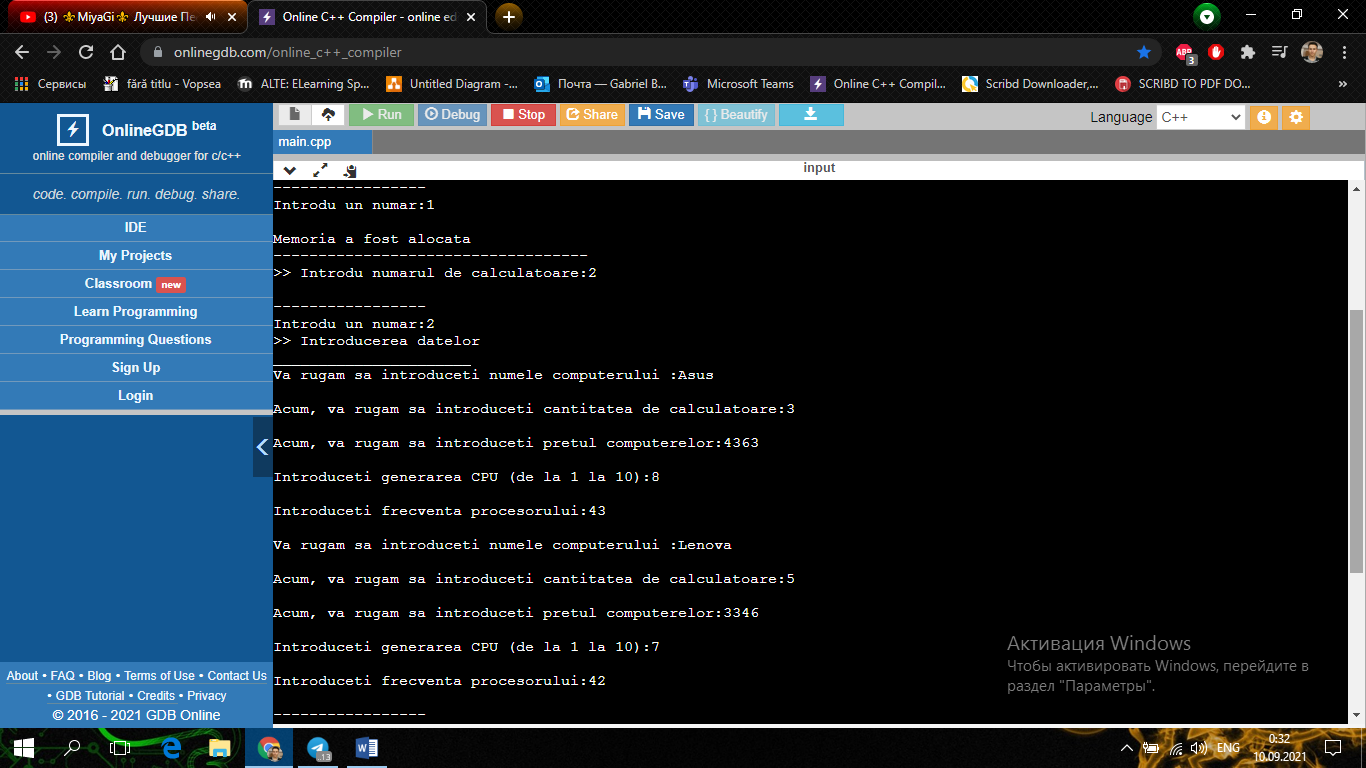
}

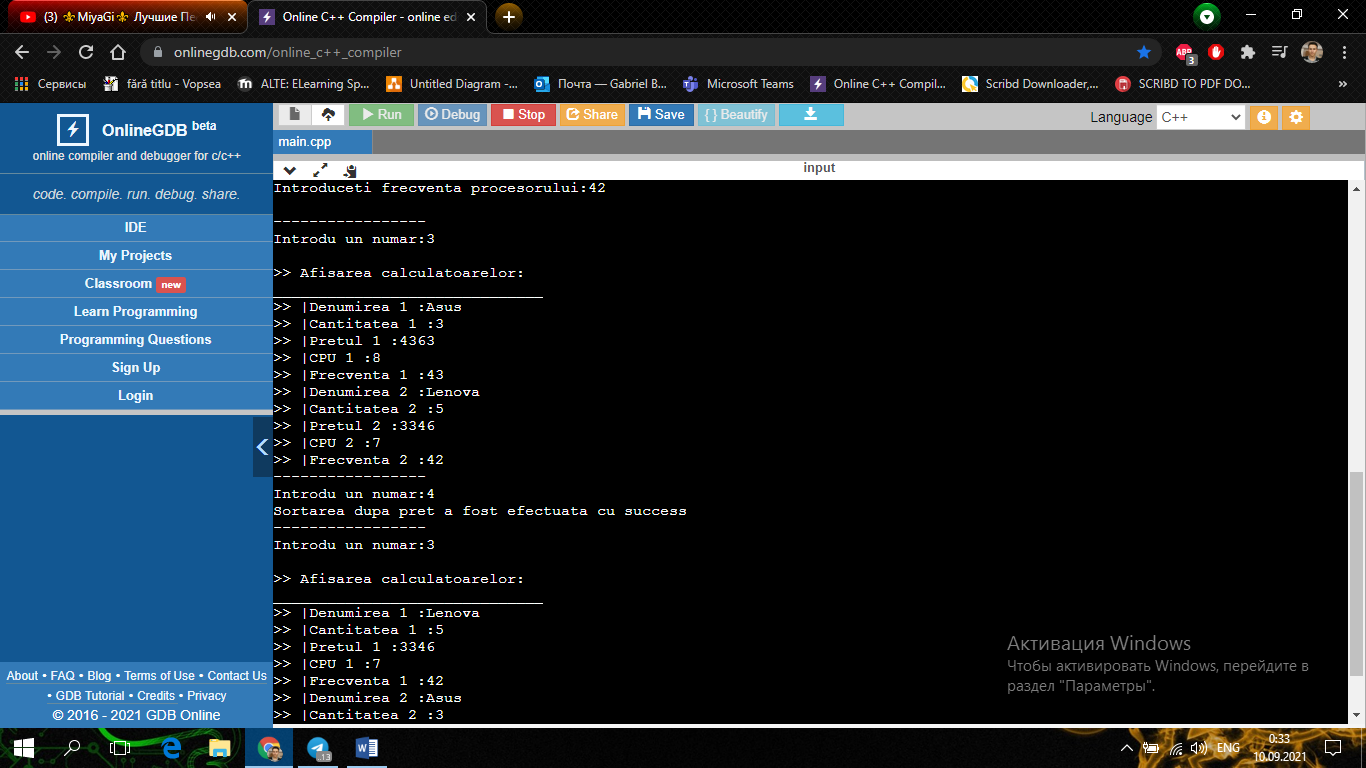
return 0;

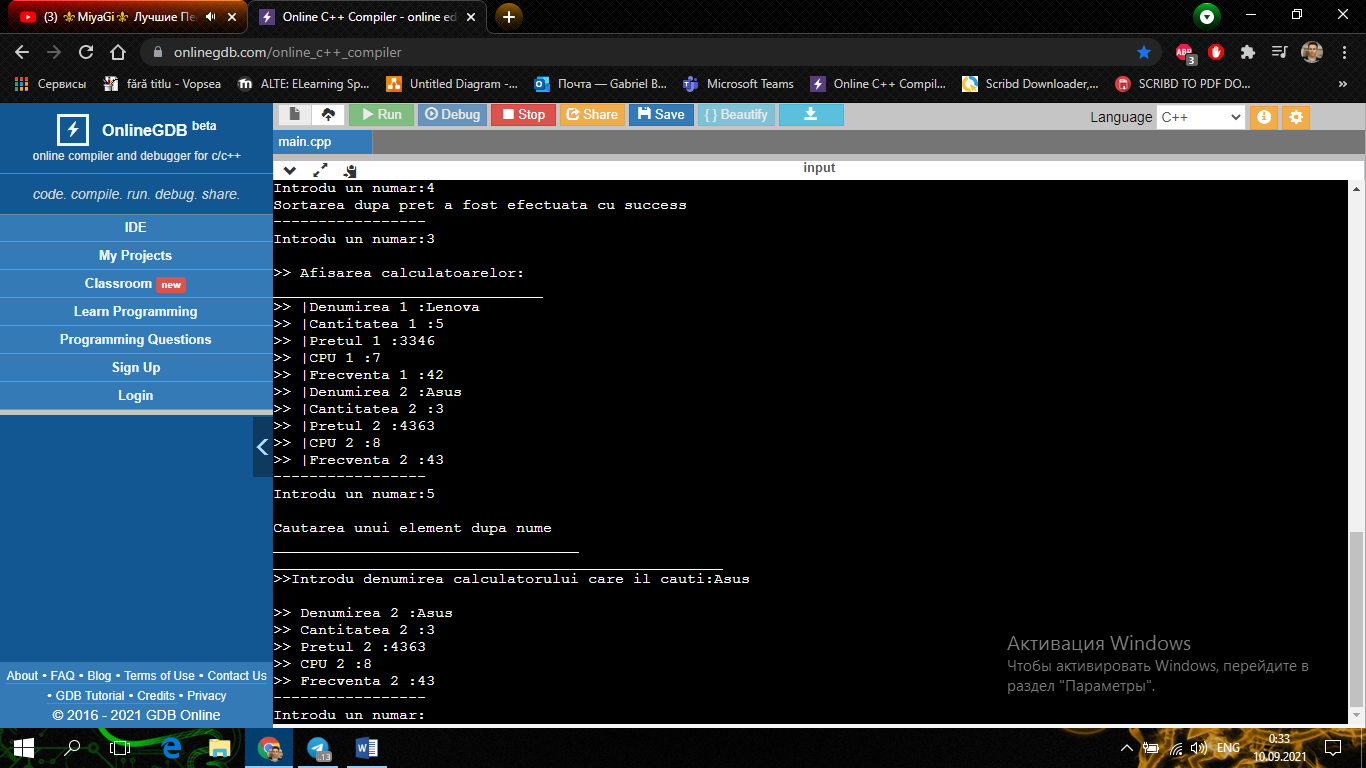
}

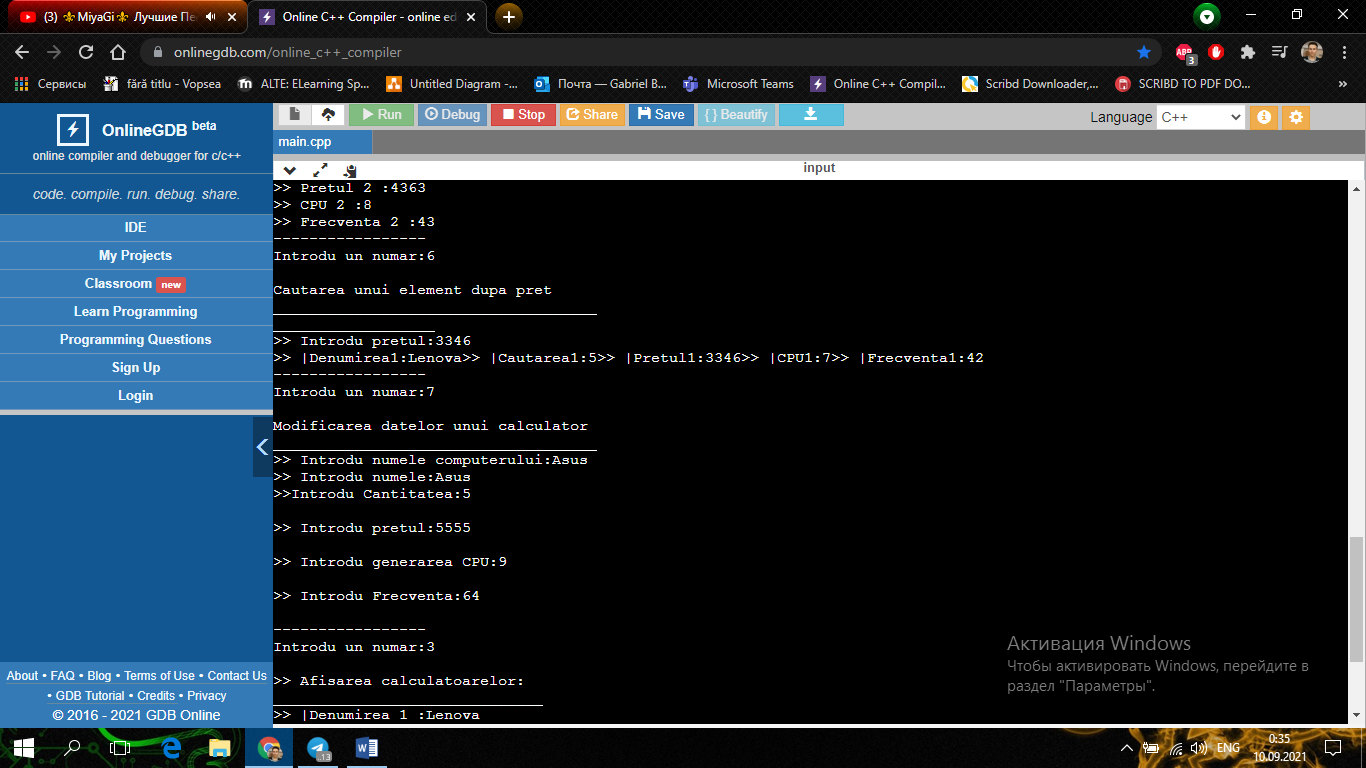
**Executia:**

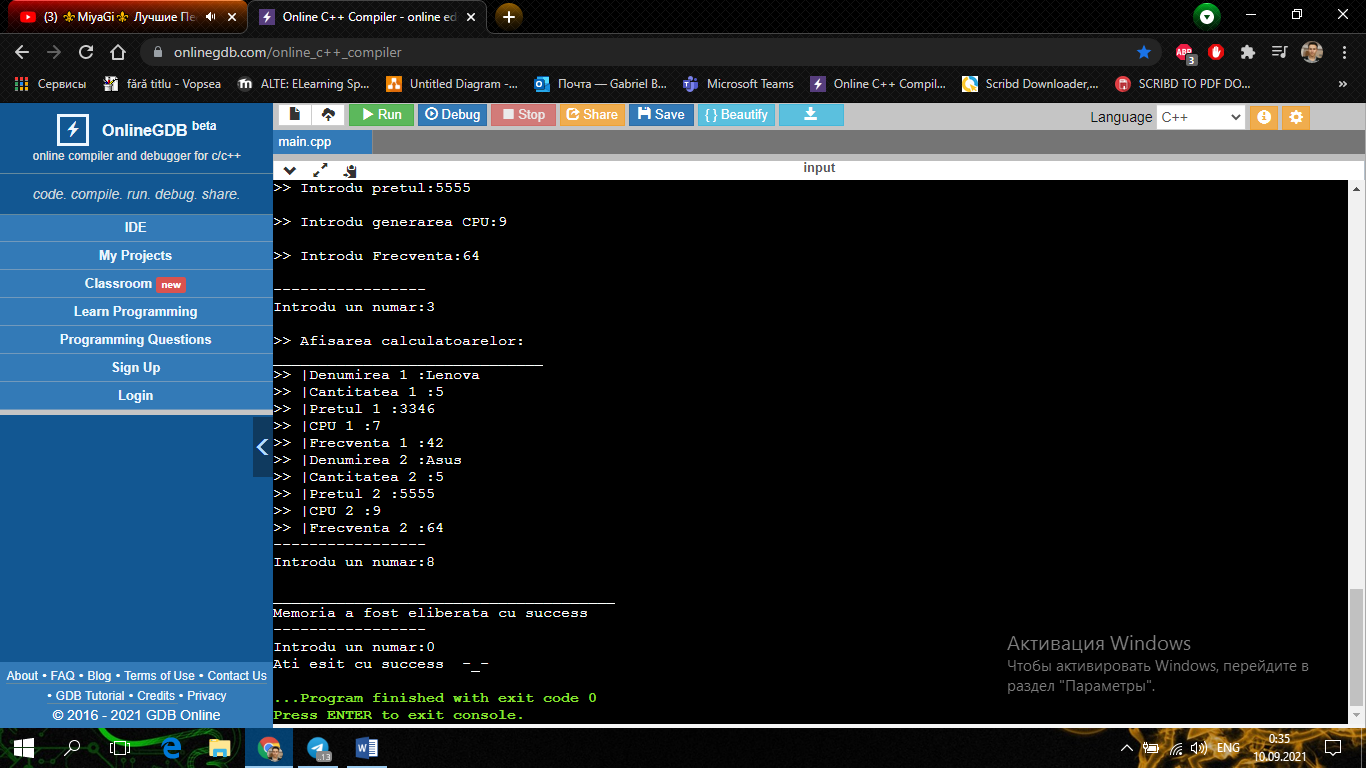












**Concluzie:**

Pe parcursul elaborării acestei lucrări de laborator eu am studiat programarea prin abstractizarea datelor, regulilor de definire şi utilizare a structurilor de date si cum se creaza variabilelor de tip structură si accesarea componentelor unei structure.