ЗВІТ

про виконання лабораторної роботи № 9.2

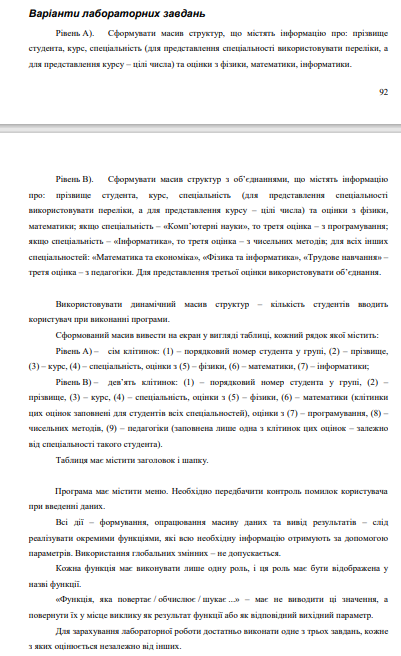
«Впорядкування та бінарний пошук в масиві структур»

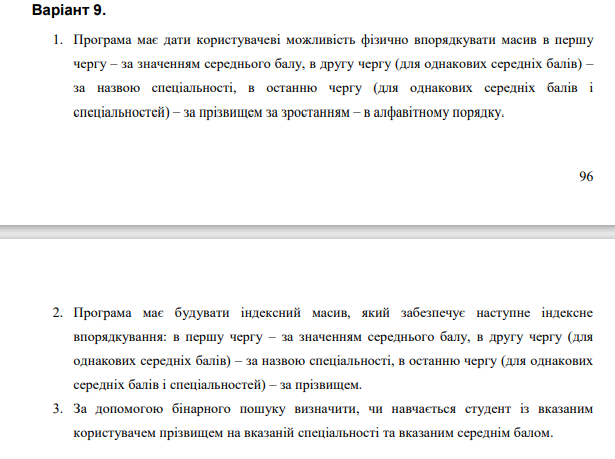
з дисципліни

«Алгоритмізація та програмування»

студента групи ІК-11

Зробка Івана Михайловича





//Варіант 9.

/\*1. Програма має дати користувачеві можливість фізично впорядкувати масив в першу

чергу – за значенням середнього балу, в другу чергу (для однакових середніх балів) –

за назвою спеціальності, в останню чергу (для однакових середніх балів і

спеціальностей) – за прізвищем за зростанням – в алфавітному порядку.

2. Програма має будувати індексний масив, який забезпечує наступне індексне

впорядкування: в першу чергу – за значенням середнього балу, в другу чергу (для

однакових середніх балів) – за назвою спеціальності, в останню чергу (для однакових

середніх балів і спеціальностей) – за прізвищем.

3. За допомогою бінарного пошуку визначити, чи навчається студент із вказаним

користувачем прізвищем на вказаній спеціальності та вказаним середнім балом.\*/

#include <iostream>

#include <iomanip>

#include <string>

#include <Windows.h>

using namespace std;

enum Group { COMPUTERSINCE, INFORMATIC, MATHANDECONOMIC, FISICSNDINFORMATIC, TUTISIONWORK };

string Namegrup[] = { "комп. науки","Інформатика","Мат. та економіка","Фізика та інф.","«Трудове навчання" };

struct student

{

string prizv{};

unsigned curs{};

Group group;

int firstmark;

int secondmark;

union

{

int prog;

int chmesod;

int workjob;

};

};

void Create(student\* p, const int N);

void Print(student\* p, const int N);

void Sort(student\* p, const int N,float arr[]);

void SerMarkFun(const int N, student\* p, float\* arr);

int BinSearch(student\* p, const int N, const string prizv, float arr[], Group group, float& sermark);

int\* indexSort(student\* p, const int N,float \*arr);

void indexPrint(student\* p, const int N, int \*a);

int main()

{ // забезпечення відображення кирилиці:

SetConsoleCP(1251); // встановлення сторінки win-cp1251 в потік вводу

SetConsoleOutputCP(1251); // встановлення сторінки win-cp1251 в потік виводу

Group group;

int iGroup;

float SerMark;

float& serMark = SerMark;

string prizv;

int N,m;

cout << "Введіть N: "; cin >> N;

m = N;

student\* p = new student[N];

float\* ArrSerBal = new float[m];

//створюем масив для запису передньго значення оцінок

int menuItem;

int found;

//дл запису і виведення порядкового номеру визначеного студента

do {

cout << endl << endl << endl;

cout << "Виберіть дію:" << endl << endl;

cout << " [1] - введення даних з клавіатури" << endl;

cout << " [2] - вивід даних на екран" << endl;

cout << " [3] - бінарний пошук працівника за посадою та прізвищем" << endl;

cout << " [4] - фізизчне впорядкування даних" << endl;

cout << " [5] - індексне впорядкування даних" << endl;

cout << " [0] - вихід та завершення роботи програми" << endl << endl;

cout << "Введіть значення: "; cin >> menuItem;

cout << endl << endl << endl;

switch (menuItem)

{

case 1:

Create(p, N);

SerMarkFun(N, p, ArrSerBal);

break;

case 2:

Print(p, N);

break;

case 3:

cout << " Введіть ключі пошуку " << endl;

cout << " група(0 - компютерні науки, 1 - інформатика, 2 - математика і економіка,";

cout << " 3 - фізика і математика, 4 - трудове навчання) : ";

cin >> iGroup; group = (Group)iGroup;

cin.get();

cin.sync();

cout << "Введіть прізвище :"; getline(cin, prizv); cout << endl;

cout << "введіть середній бал студента (має бури 5 цифер після коми ) : "; cin >> serMark;

if (found = BinSearch(p, N, prizv, ArrSerBal, group, serMark))

cout << "Знайдено працівника в позиції " << found + 1 << endl;

else

cout << "Шуканого працівника не знайдено" << endl;

break;

case 4:

Sort(p, N,ArrSerBal);

break;

case 5:

indexPrint(p, N, indexSort(p, N,ArrSerBal));

case 0:

break;

default:

cout << "Ви ввели помилкове значення! "

"Слід ввести число - номер вибраного пункту меню" << endl;

}

} while (menuItem != 0);

return 0;

}

void SerMarkFun(const int N, student\* p, float\* arr)

{

for (int i = 0; i < N; i++)

{

arr[i] = (p[i].firstmark + p[i].secondmark + (p[i].chmesod || p[i].prog || p[i].workjob)) / 3;

}

}

int BinSearch(student\* p, const int N, const string prizv, float arr[],Group group,float &sermark)

{

int L = 0, R = N - 1, m;

do {

m = (L + R) / 2;

if (p[m].prizv == prizv && p[m].group == group)

{

for (int i = 0; i < N; i++)

{

if (arr[i] == sermark)

return m;

}

}

if ((p[m].group < group) || (p[m].group == group && p[m].prizv < prizv) ||

(p[m].group == group && p[m].prizv == prizv))

for (int i = 0; i < N; i++)

{

if (arr[i] == sermark)

L = m + 1;

}

else

R = m - 1;

} while (L <= R);

}

void Sort(student\* p, const int N,float arr[])

{

student tmp;

for (int i0 = 0; i0 < N-1; i0++)

{

for (int i1 = 0; i1 < N - 1 - i0; i1++)

{

if ((arr[i1]>arr[i1 + 1])

||

(arr[i1] == arr[i1 + 1]&& p[i1].group > p[i1 + 1].group)

||

((arr[i1] == arr[i1 + 1])&& (p[i1].group == p[i1 + 1].group)&& (p[i1].prizv == p[i1].prizv)))

{

tmp = p[i1];

p[i1] = p[i1 + 1];

p[i1 + 1] = tmp;

}

}

}

}

void Create(student\* p, const int N)

{

int groupN;

for (int i = 0; i < N; i++)

{

cout << "Працівник № " << i + 1 << ":" << endl;

cin.get(); // очищуємо буфер клавіатури – бо залишаються символи

cin.sync(); // "кінець рядка", які не дають ввести наступний літерний рядок

cout << " прізвище: "; getline(cin, p[i].prizv);

cout << " курс "; cin >> p[i].curs;

cout << " група (0 - компютерні науки, 1 - інформатика, 2 - математика і економіка, 3 - фізика і математика, 4 - трудове навчання): ";

cin >> groupN;

cout << " оцінка з фізики "; cin >> p[i].firstmark;

cout << " оцінка з математики "; cin >> p[i].secondmark;

p[i].group = (Group)groupN;

switch (p[i].group)

{

case COMPUTERSINCE:

cout << "оцінка з програмування"; cin >> p[i].prog;

break;

case INFORMATIC:

cout << "оцінка з чисельних методів "; cin >> p[i].chmesod;

break;

case MATHANDECONOMIC:

case FISICSNDINFORMATIC:

case TUTISIONWORK:

cout << "оцінка з педагогіки "; cin >> p[i].workjob;

break;

}

cout << endl;

}

}

void Print(student\* p, const int N)

{

cout << "=================================================================================="

<< endl;

cout << "| № | Прізвище | Курс | група | 1 оцінка | 2 оцінка | 3 оцінка |"

<< endl;

cout << "----------------------------------------------------------------------------------"

<< endl;

for (int i = 0; i < N; i++)

{

cout << "| " << setw(1) << right << i + 1 << " ";

cout << "| " << setw(15) << left << p[i].prizv

<< "| " << setw(4) << right << p[i].curs << " "

<< "| " << setw(18) << left << Namegrup[p[i].group];

switch (p[i].group)

{

case COMPUTERSINCE:

cout << "| " << p[i].firstmark << " |" << " " << p[i].secondmark << " |" << p[i].secondmark << " |";

cout << endl;

break;

case INFORMATIC:

cout << "| " << p[i].firstmark << " |" << " " << p[i].secondmark << " |" << p[i].chmesod << " |";

cout << endl;

break;

case MATHANDECONOMIC:

case FISICSNDINFORMATIC:

case TUTISIONWORK:

cout << "| " << p[i].firstmark << " |" << " " << p[i].secondmark << " |" << p[i].workjob << " |";

cout << endl;

break;

}

}

cout << "=================================================================================="

<< endl;

cout << endl;

}

int\* indexSort(student\* p, const int N,float \*arr)

{

int\* I = new int[N];

for (int i = 0; i < N; i++)

{

I[i] = i;

}

int i, j, value;

for (int i = 1; i < N; i++)

{

value = I[i];

for (j = i-1; j>=0 &&

((arr[I[j]] > arr[value])

||

(arr[I[j]] == arr[value] && p[I[j]].group > p[value].group)

||

((arr[I[j]] == arr[I[j]]) && (p[I[j]].group == p[value].group) && (p[I[j]].prizv == p[I[j]].prizv)))

; j--)

{

I[j + 1] = I[j];

}

I[j + 1] = value;

}

return I;

}

void indexPrint(student\* p, const int N, int \*a)

{

cout << "=================================================================================="

<< endl;

cout << "| № | Прізвище | Курс | група | 1 оцінка | 2 оцінка | 3 оцінка |"

<< endl;

cout << "----------------------------------------------------------------------------------"

<< endl;

for (int i = 0; i < N; i++)

{

cout << "| " << setw(1) << right << i + 1 << " ";

cout << "| " << setw(15) << left << p[a[i]].prizv

<< "| " << setw(4) << right << p[a[i]].curs << " "

<< "| " << setw(18) << left << Namegrup[p[a[i]].group];

switch (p[a[i]].group)

{

case COMPUTERSINCE:

cout << "| " << p[a[i]].firstmark << " |" << " " << p[a[i]].secondmark << " |" << p[a[i]].secondmark << " |";

cout << endl;

break;

case INFORMATIC:

cout << "| " << p[a[i]].firstmark << " |" << " " << p[a[i]].secondmark << " |" << p[a[i]].chmesod << " |";

cout << endl;

break;

case MATHANDECONOMIC:

case FISICSNDINFORMATIC:

case TUTISIONWORK:

cout << "| " << p[a[i]].firstmark << " |" << " " << p[a[i]].secondmark << " |" << p[a[i]].workjob << " |";

cout << endl;

break;

}

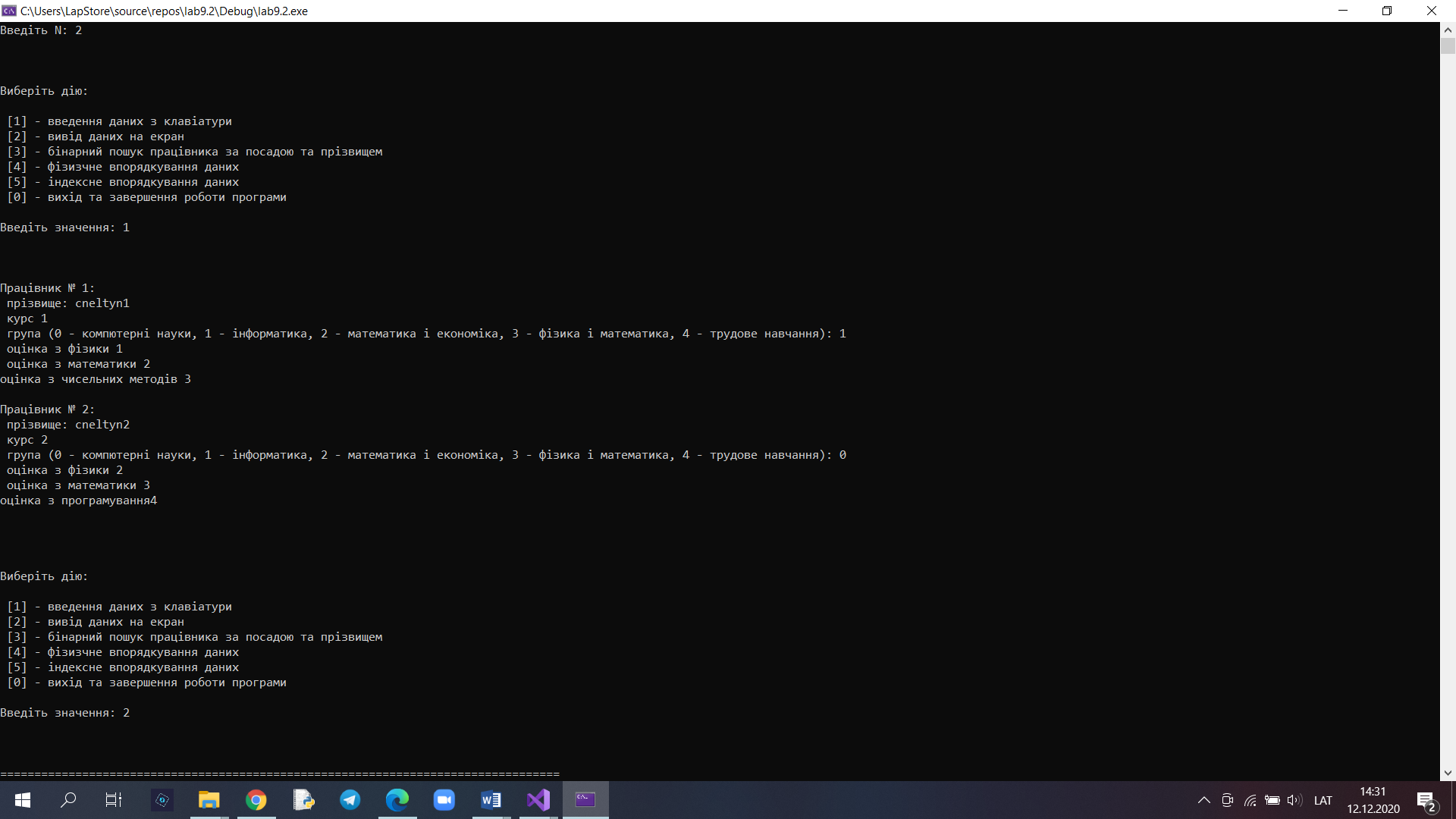
}

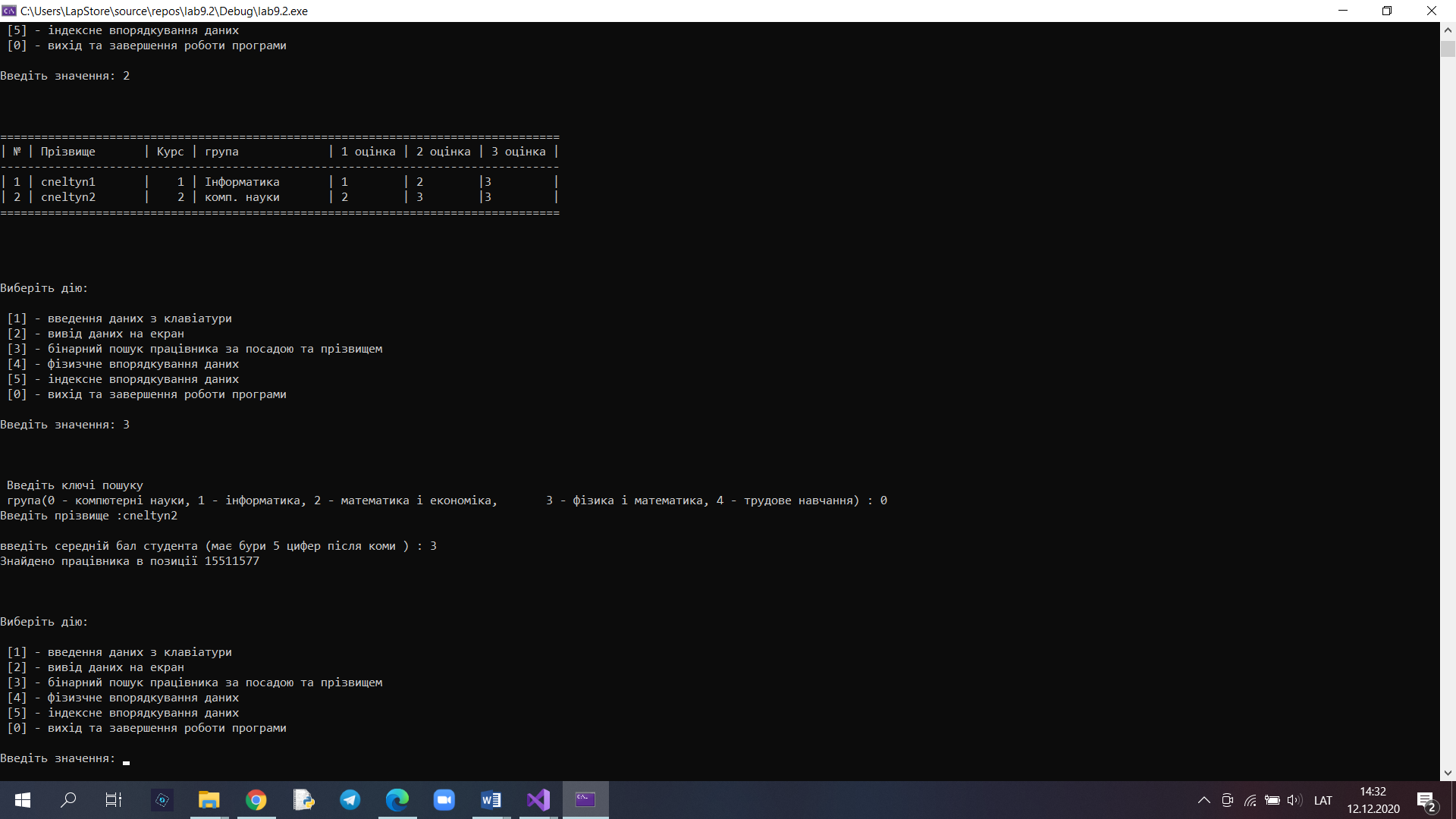
cout << "=================================================================================="

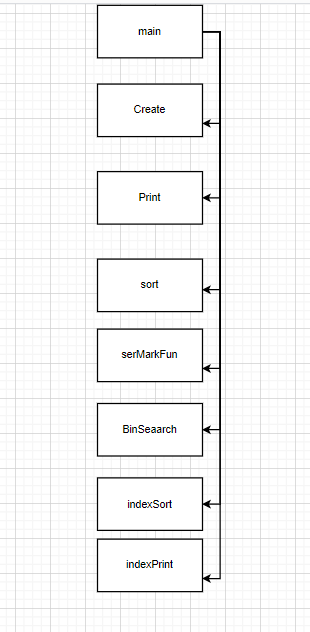
<< endl;

cout << endl;

}







Посилання на GitHub

https://github.com/ivanzrobok/lab9.2

Висновок : навчився сортувати масив структур та виконувати пошук по ньому.