ЗВІТ

Про виконання лабораторної роботи № 10.2 A

«Впорядкування та бінарний пошук в масиві структур»

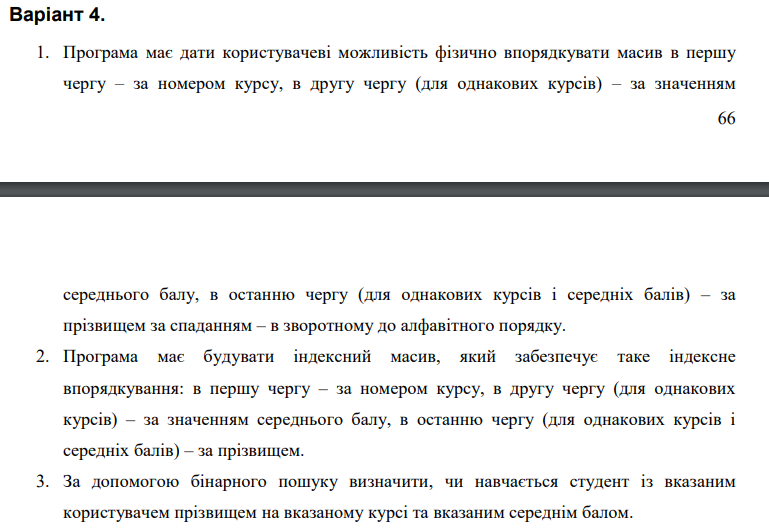
З дисципліни

“Алгоритмізація та програмування”

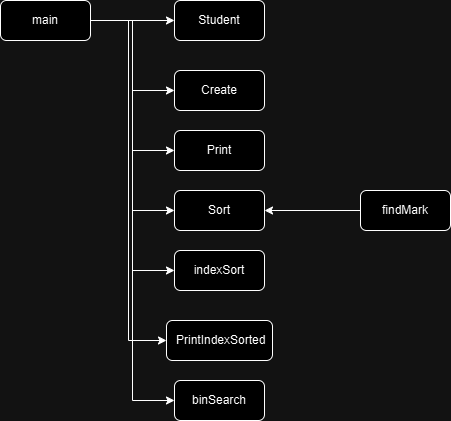
Студента групи РІ-11

Ганина Дмитра Юрійовича

**Умова завдання**

****

**Структура:**

****

**Текст програми:**

#include <iostream>

#include <string>

#include <Windows.h>

#include <iomanip>

using namespace std;

enum Spec { CS, IT, ME, PhIT, TR };

string strSpec[] = { "Комп'ютернi науки", "Iнформатика", "Математика та економiка", "Фiзика та iнформатика", "Трудове навчання" };

struct Student

{

string lname;

int kurs;

Spec spec;

int physics;

int math;

int informatik;

};

void Create(Student\* s, int amSt) {

int sp;

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

cout << "Прiзвище: ";

cin >> s[i].lname;

cout << "Курс: ";

cin >> s[i].kurs;

cout << "Спецiальнiсть (0 - Комп'ютернi науки, 1 - Iнформатика, 2 - Математика та економiка, 3 - Фiзика та iнформатика, 4 - Трудове навчання) ";

cin >> sp;

s[i].spec = (Spec)sp;

cout << "Оцiнка з фiзики: ";

cin >> s[i].physics;

cout << "Оцiнка з математики: ";

cin >> s[i].math;

cout << "Оцiнка з інформатики: ";

cin >> s[i].informatik;

cout << endl;

}

}

void Print(Student\* s, int amSt)

{

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

<< endl;

cout << "| № | Прiзвище | Курс | Спецiальнiсть | Фiзика | Математика | Інформатика |"

<< endl;

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

<< endl;

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

{

cout << "| " << setw(2) << right << i + 1 << "";

cout << "| " << setw(14) << left << s[i].lname

<< "| " << setw(5) << right << s[i].kurs

<< "| " << setw(23) << left << strSpec[s[i].spec]

<< "| " << setw(7) << s[i].physics

<< "| " << setw(11) << right << s[i].math

<< "| " << setw(7) << right << s[i].informatik;

cout << endl;

}

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

<< endl;

}

double findMark(Student\* p, const int N) {

return ((p[N].math + p[N].physics + p[N].informatik) / 3.0);

}

// Фізичне впорядкування

void Sort(Student\* p, const int N)

{

Student tmp;

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

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

if ((p[i1].kurs > p[i1 + 1].kurs) ||

(p[i1].kurs == p[i1 + 1].kurs && findMark(p, i1) < findMark(p, i1 + 1)) ||

(p[i1].kurs == p[i1 + 1].kurs && findMark(p, i1) == findMark(p, i1 + 1) &&

p[i1].lname < p[i1 + 1].lname))

{

tmp = p[i1];

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

p[i1 + 1] = tmp;

}

}

// Індексне впорядкування

int\* indexSort(Student\* s, const int N)

{

int\* P = new int[N];

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

P[i] = i;

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

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

if ((s[P[i1]].kurs > s[P[i1 + 1]].kurs) ||

(s[P[i1]].kurs == s[P[i1 + 1]].kurs && findMark(s, P[i1]) < findMark(s, P[i1 + 1])) ||

(s[P[i1]].kurs == s[P[i1 + 1]].kurs && findMark(s, P[i1]) == findMark(s, P[i1 + 1]) &&

s[P[i1]].lname < s[P[i1 + 1]].lname))

{

int tmp = P[i1];

P[i1] = P[i1 + 1];

P[i1 + 1] = tmp;

}

return P;

}

// Виведення індексно-впорядкованого масиву

void PrintIndexSorted(Student\* s, int\* P, const int N)

{

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

<< endl;

cout << "| № | Прiзвище | Курс | Спецiальнiсть | Фiзика | Математика | Інформатика |"

<< endl;

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

<< endl;

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

{

cout << "| " << setw(2) << right << i + 1 << "";

cout << "| " << setw(14) << left << s[P[i]].lname

<< "| " << setw(5) << right << s[P[i]].kurs

<< "| " << setw(23) << left << strSpec[s[P[i]].spec]

<< "| " << setw(7) << s[P[i]].physics

<< "| " << setw(11) << right << s[P[i]].math

<< "| " << setw(7) << right << s[P[i]].informatik;

cout << endl;

}

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

<< endl;

}

// Бінарний пошук

int binSearch(Student\* p, const int N, const string prizv, int const kurs, const double avr)

{

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

{

if (p[i].lname == prizv && p[i].kurs == kurs && findMark(p, i) == avr) {

return i;

}

}

return -1;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int amSt;

cout << "Кількість студентів: ";

cin >> amSt;

Student\* s = new Student[amSt];

int menuItem;

do {

cout << endl << endl;

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

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

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

cout << " [3] - фізичне впорядкування даних" << endl;

cout << " [4] - індексне впорядкування та вивід даних" << endl;

cout << " [5] - бінарний пошук студента" << endl;

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

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

cout << endl;

switch (menuItem)

{

case 1:

Create(s, amSt);

break;

case 2:

Print(s, amSt);

break;

case 3:

Sort(s, amSt);

cout << "Дані впорядковано!" << endl;

break;

case 4:

{

int\* P = indexSort(s, amSt);

PrintIndexSorted(s, P, amSt);

delete[] P;

break;

}

case 5:

{

string prizv;

int kurs, avrGrade;

cout << "Введіть прізвище: "; cin >> prizv;

cout << "Введіть курс: "; cin >> kurs;

cout << "Введіть середню оцінку: "; cin >> avrGrade;

int found = binSearch(s, amSt, prizv, kurs, avrGrade);

if (found != -1)

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

else

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

break;

}

case 0:

break;

}

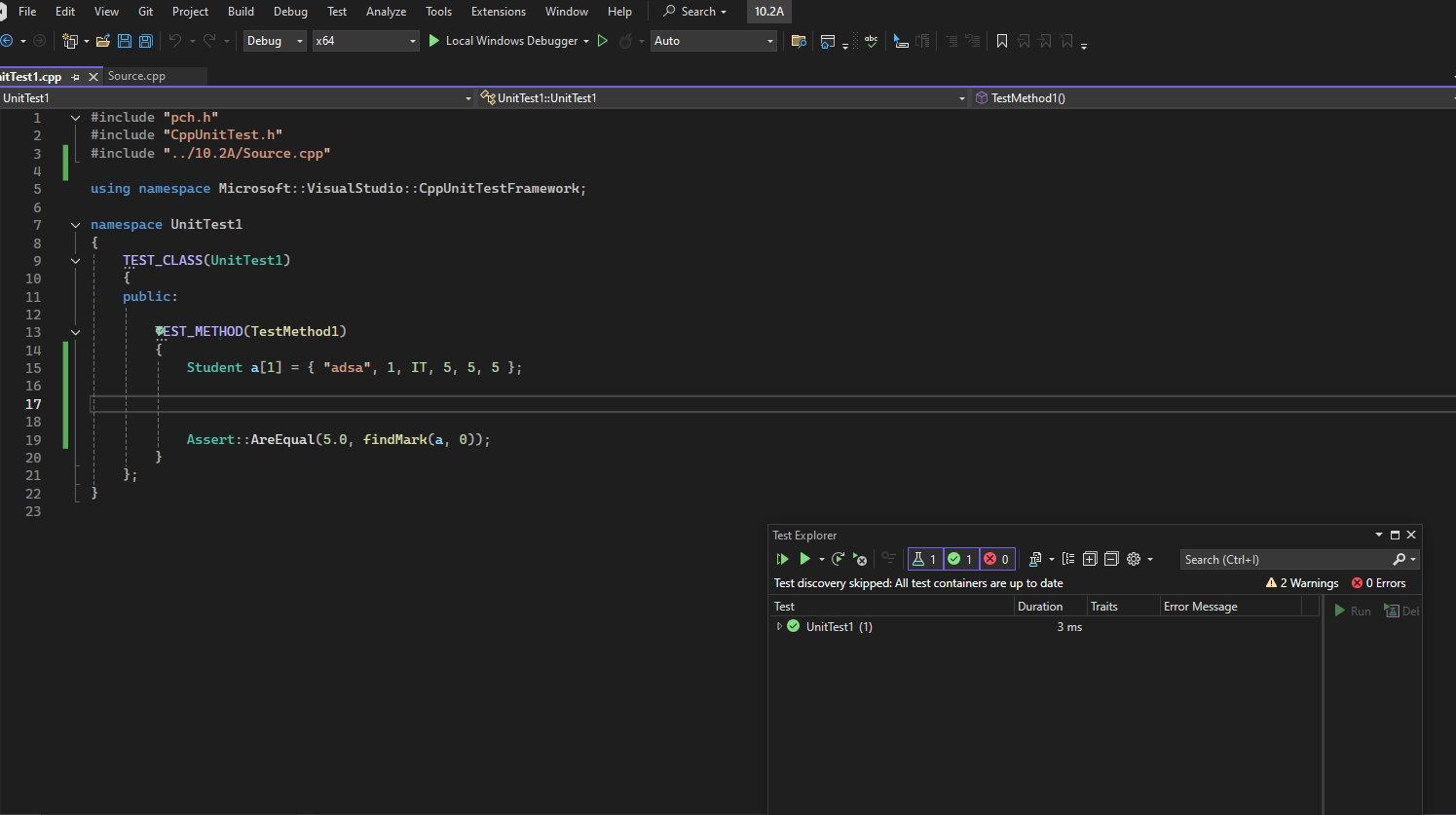
} while (menuItem != 0);

delete[] s;

return 0;

}

**Unit test:**

****

**Посилання на git-репозиторій з проектом:**

<https://github.com/HanynDmytro/Lab10.2A>

**Висновки:**

Під час виконання лабораторної роботи я навчився впорядковувати масив структур з об’єднаннями, здійснювати фізичне та індексне впорядкування і здійснювати бінарний пошук у фізично чи індексно впорядкованому масиві