 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Комп’ютерний практикум №15**

з дисципліни «Алгоритмізація та програмування»

на тему: «Структури»

Варіант №17

**Виконав:**

студент гр. БС-81

Сєров О. В.

**Перевірив:**

доцент каф. БМК

к.т.н. Алхімова С.М.

Зараховано від \_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис викладача)

Київ-2019

**Завдання:**

1. Ознайомитися з теоретичними відомостями щодо використання структур в мові C++ та способів доступу до їх елементів.
2. Побудувати блок-схему алгоритму для вирішення задачі відповідно до свого варіанту.
3. Створити статичний масив структур, що містить інформацію відповідно до свого варіанту завдання та розробити програмний застосунок, що включає його ініціалізацію, обробку та виведення на екран результатів (поля структури, що є рядками, задавати статичними масивами символів, ініціалізацію масиву структур провести в коді програми, глобальні змінні не використовувати, функція main має бути призначена тільки для виклику функцій користувача):

***Відомості про результати півфіналу конкурсу містять наступну інформацію: прізвище, ім’я, адреса, секція, набраний бал. Знайти прохідний бал у фінал конкурсу по кожній із секцій, якщо кількість учасників фіналу відома (кількість вводиться з клавіатури), і скласти списки учасників, які пройшли у фінал по кожній із секцій.***

1. Скласти і захистити звіт по роботі.

**🞏 Комп’ютерний практикум без зауважень**

**🞏 Комп’ютерний практикум має зауваження:**

**🞏 несвоєчасний захист**

**🞏 присутні зауваження до блок-схеми:**

**🞏 блок-схема не відповідає коду**

**🞏 в блок-схемі присутній код**

**🞏 виконані не за стандартом:**

**🞏 блок умови 🞏 визначений процес (функція)**

**🞏 оператор вибору 🞏 перехід**

**🞏 цикл 🞏 розміри блоків**

**🞏 інші зауваження:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**🞏 присутні зауваження до коду:**

**🞏 задача завдання вирішена хибно**

**🞏 код програми не компілюється**

**🞏 використано глобальні змінні**

**🞏 типи даних визначені хибно**

**🞏 недостатня декомпозиція на функції користувача**

**🞏 функція main містить лише виклик іншої функції**

**🞏 статичні змінні при роботі з масивами**

**🞏 оформлення коду**

**🞏 присутні зайві символи «{» та «}»**

**🞏 інші зауваження:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**🞏 результати виконання програми на рисунках не відповідають коду**

**🞏 невірні відповіді на запитання:**

**🞏 №1 🞏 №2 🞏 №3 🞏 №4 🞏 №5**

**🞏 №6 🞏 №7 🞏 №8 🞏 №9 🞏 №10**

**🞏 незнання теоретичного матеріалу**

**🞏 маються інші зауваження:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Блок-схема:**







**Код:**

#include <iostream>

#include <conio.h>

using namespace std;

struct info

{

char surname[16];

char name[16];

char address[32];

char section[16];

int score;

};

int section\_compare(const void\* sec1, const void\* sec2)

{

return strcmp(((struct info\*)sec1)->section, ((struct info\*)sec2)->section);

}

int score\_compare(const void\* score1, const void\* score2)

{

return (((struct info\*)score1)->score - ((struct info\*)score2)->score);

}

void print\_info(const struct info\* competitor)

{

cout << "\nCompetitor :\nSurname : " << competitor->surname

<< "\nName : " << competitor->name

<< "\nAddress : " << competitor->address

<< "\nSection : " << competitor->section

<< "\nScore : " << competitor->score << endl;

}

void finalists\_info(info\* array, int finalists\_number = 1)

{

qsort(array, 12, sizeof(info), section\_compare);

int j = 0;

for (int i = 0; i < 12; i++)

{

if (!strcmp(array[i].section, array[i + 1].section))

continue;

else

{

if (j)

{

info\* best\_competitors = new info[i - j];

for (int k = i - j; k > 0; k--)

best\_competitors[k-1] = array[j + k];

qsort(best\_competitors, i - j, sizeof(info), score\_compare);

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

cout << "\n\nThe passing score in " << array[i].section << " : "

<< best\_competitors[i - j - finalists\_number].score << endl

<< "Applause to finalists at this discipline :\n";

for (int k = i - j; k > i - j - finalists\_number; k--)

print\_info(&best\_competitors[k-1]);

delete[]best\_competitors;

}

else

{

info\* best\_competitors = new info[i - j + 1];

for (int k = i - j; k >= 0; k--)

best\_competitors[k] = array[j + k];

qsort(best\_competitors, i - j + 1, sizeof(info), score\_compare);

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

cout << "\n\nThe passing score in " << array[i].section << " : "

<< best\_competitors[i - j - finalists\_number + 1].score << endl

<< "Applause to finalists at this discipline :\n";

for (int k = i - j; k > i - j - finalists\_number; k--)

print\_info(&best\_competitors[k]);

delete[]best\_competitors;

}

}

j = i;

}

}

void final()

{

info competitors[12] = {

{"Malaev", "Andrew", "avenue Grand Lovetto 43", "Tennis", 15},

{"Yang", "Larry", "st. Rose park 14", "Tennis", 16},

{"Travis", "Dorry", "avenue Adams Dene 67", "Cybersport", 145},

{"Gibson", "Daisy", "alley of Wonder 54/1", "Tennis", 18},

{"Bridges", "Endy", "st. Parsonage Park Drive 101", "Hackathon", 1244},

{"Smell", "Alexander", "alley Robertson 51", "Cybersport", 125},

{"Jackson", "Gordon", "st. Slowy 221", "Hackathon", 2412},

{"Stickson", "Richard", "st. Blackthorn Common 42", "Cybersport", 135},

{"Garrison", "Frederick", "alley Wesker Place 64", "Cybersport", 120},

{"Mescudi", "Chris", "alley Beaulieu Heights 51/2", "Hackathon", 1121},

{"Dallas", "Morgan", "avenue Parkham Mead 22", "Hackathon", 2445},

{"Valakas", "Glad", "st. Roflan 228", "Tennis", 17}

};

int finalists\_number = 0;

cout << "}------------------{Big tournament began!}------------------{\n\n"

<< "Enter the number of finalists in every section : ";

cin >> finalists\_number;

finalists\_info(competitors, finalists\_number);

}

void main()

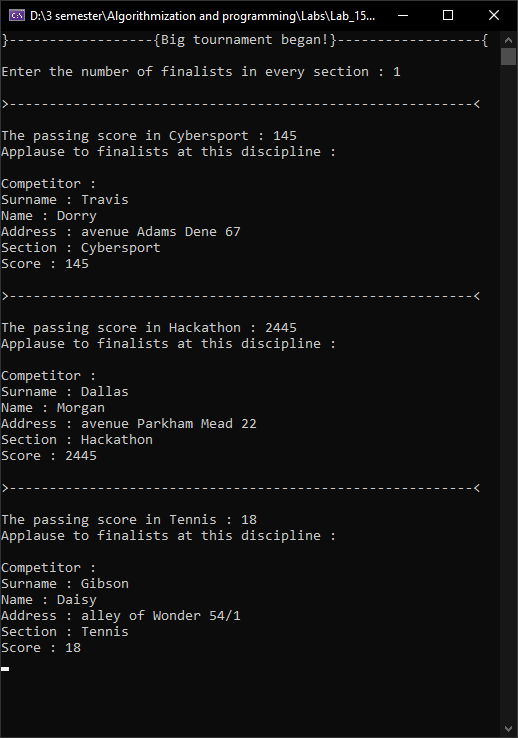
{

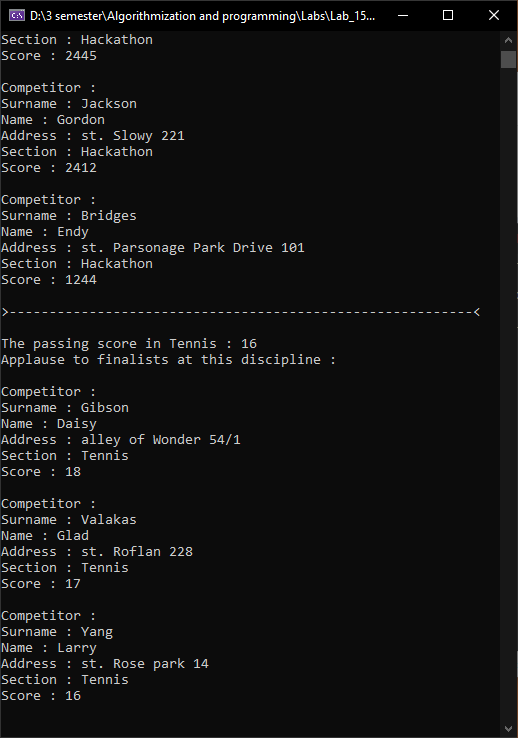
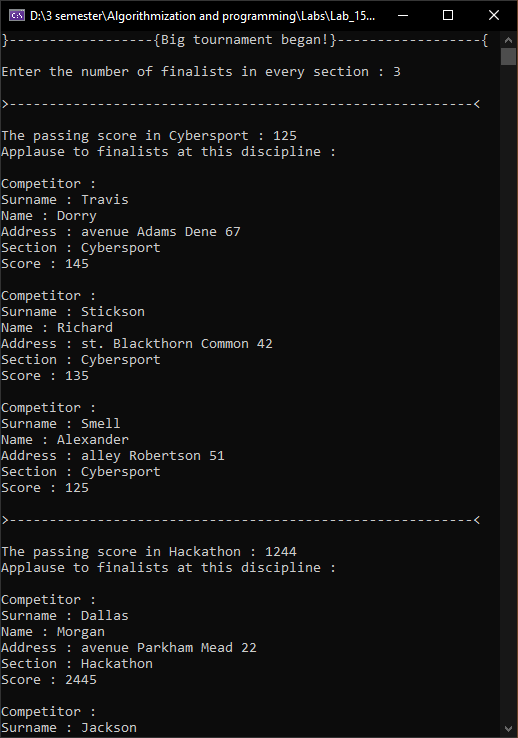
final();

\_getch();

}

**Результати:**

****

****

**Контрольні питання:**

1. ***Що таке структура, для чого вона призначена?***

Структура - це тип даних користувача, в якому під одним ім’ям об’єднані дані різних типів. Їх використання:

* + Полегшує написання і розуміння програм;
  + Допомагає згрупувати дані, що об’єднують якесь загальне поняття;
  + Дозволяє групу пов’язаних між собою змінних використовувати як безліч окремих елементів і водночас як єдине ціле.

1. ***Що таке вкладена структура?***

Вкладена структура – структура, що є одним із типів даних іншої структури.

1. ***Способи оголошення нових структурних змінних.***

* Структурний шаблон з ім'ям:

*struct ім’я\_шаблону {<опис полів структури> };*

* Структурний шаблон з ім'ям і списком змінних:

*struct ім’я\_шаблону {<опис полів структури>} <Список змінних>;*

* Структурний шаблон без імені:

*struct {<опис полів структури>} <Список змінних>;*

де struct – ключеве слово, яке оголошує структуру, ім’я шаблону – ім’я структурного типу, а в якості полів задаються змінні.

1. ***Як ініціалізувати поле структури, яке є масивом?***

example\_struct[3] = {{0,1}, {1,2}, {2,0}}

1. ***Як оголосити покажчик на структуру?***
   * Використовуючи операції розіменування «\*»
   * Використовуючи покажчик «->»
2. ***Як ініціалізувати покажчик на структуру?***

Так як і для звичайного типу:

truct example\_struct \*ptr;

ptr = &var

1. ***Які операції використовуються для доступу до елементів структури, в чому їх різниця?***

До полів структури можна звергатися через складене ім’я, яке містить ім’я структури, операції доступу до полів «.» або «->» та ім’я поля. Формат звернення: struct\_name.field або struct\_name->field. Операція доступу «.» потребує, щоб зліва від неї знаходився ідентифікатор змінної, а операція доступу «->» працює з покажчиками.

1. ***Яким чином передаються структури до функції?***
   * + - void foo(example\_struct \*ptr) foo(&var)
       - void foo(example\_struct &ref) foo(var)
       - void foo(example\_struct val) foo(var)
2. ***Чи можна передавати в функцію масив структур?***

Так, у функцію можна передавати масив структур, так як цей масив не відрізняється від звичних масивів.

e.g. void finalists\_info(info\* array)

1. ***Які проблеми можуть виникнути під час копіювання змінних структурного типу одна в одну?***

Дані типу простих структур можна присвоювати один одному тільки тоді, якщо усі члени обох структур мають числові типи. При цьому, в операції присвоювання обидва операнда повинні бути структурного типу, тобто відбудеться копіювання. Якщо розміри структур відрізняються – копіюється кількість байт, що відповідає меншому розміру.