**Лабораторна робота №15 Написання програм з використанням функцій та файлів.**

**На період** карантину в дистанційній формі навчання потрібно використовувати замість **С++ Builder онлайн компілятор C++ Shell, який доступний за адресом** [**http://cpp.sh**](http://cpp.sh) (інструкція надана після тексту ЛР№3). При наявності на домашньому комп’ютері іншого компілятору С++ завдання можна виконати на ньому. Результати викласти на платформу коледжу в свій репозиторій та надсилати на електронну адресу викладача [**t.i.lumpova@gmail.com**](mailto:t.i.lumpova@gmail.com)у вигляді cpp-файлу з іменем у форматі

**<Номер групи><Номер лабораторної>[-<Номер завдання>] <Прізвищеанглійською>**

Наприклад,21-1Loban.cpp.

При відсутності можливості доступу до Інтернету текст програми набрати в Блокноті або WordPad Windows та надіслати на електронну адресу викладача

**Увага!** C++Shell не підтримує роботу з файлами.

Для роботи з файлами можна використати безкоштовний компілятор Dev-C++, який потрібно встановити на ваш комп’ютер. Скачати його можна за посиланням:

**https://sourceforge.net/projects/orwelldevcpp/**

Інструкція по роботі з компілятором надається за посиланням (але там забагато реклами та іншого сміття):

[**https://studopedia.ru/18\_64672\_pokrokove-vikonannya-programi.htm**](https://studopedia.ru/18_64672_pokrokove-vikonannya-programi.htm)

Якщо у Вас нема можливості встановити цю програму, то просто запишіть код виконання операцій роботи з файлами та надішліть на перевірку.

Іншим рішенням є надсилання поштою посилання на текст програми за URL адреси, яку надає C++Shell, вказавши в темі листа, номер групи прізвище студента та номер ПР.

В темі листа вказати, номер групи, прізвище студента та номер ПР як "ПР№8".

**Строк відсилки ЛР 04.05.2021**

**Мета**: навчитися модифікувати програми **на мові С++** в **консольному режимі** з використанням функцій та файлів**.**

**Методичні вказівки щодо організації самостійної роботи студентів**

1. Повторити Лекції 9, 10 теоретичні відомості з Практичних робіт №7,8 та з Лабораторних робіт №7 – 14, а також до цієї Лабораторної роботи.
2. Запустити середовище програмування С++ .

**Завдання.**

1. Записати програму, модифікувавши Приклад 1 з Теоретичної частини. Виконати наступні дії
   1. В першому рядку програми записати

*// ПТБД-21 Група № Прізвище*

вказавши номер своєї групи та своє прізвище.

* 1. Замість рядкових змінних використати змінні типу string.
  2. Функцію Example7 викликати без параметрів, а ім’я файлу задати літералом "Example7.dat" в оголошенні файлового потоку.
  3. Скомпілювати та виконати програму

1. Результати надсилати на електронну адресу викладача [**t.i.lumpova@gmail.com**](mailto:t.i.lumpova@gmail.com)

***Контрольні запитання для самоперевірки***.

1. Чим різниться введення-виведення файлів в С-style та С++ - стилі?
2. В якому випадку доцільно використовувати функції write(), read()?

**ТЕОРЕТИЧНІ ВІДОМОСТІ**

**Приклад 1**. Читання/запис масиву об’єктів у файл. Функції write(), read()

У прикладі використовуються функції write(), read() для роботи зі об’єктами класу BOOK, а саме:

* запис масиву об’єктів класу BOOK у файл;
* читання масиву об’єктів класу BOOK з файлу.

**#include <iostream>**

**#include <fstream>**

**#include <string.h>**

**#include <Windows.h>**

**using namespace std;**

**// Клас BOOK**

**class BOOK**

**{public:**

**char title[100]; // Назва книги**

**char author[70]; // Автор**

**int year; // Рік видання**

**float price; // вартість книги**

**};**

**// Запис масиву об’єктів в файл з допомогою функції write()**

**// читання масиву об’єктів з файлу з допомогою функції read()**

**bool Example7(const char \* filename)**

**{**

**// створити масив об’єктів**

**BOOK B[3] =**

**{**

**{ "Title-01", "Author-01", 2005, 100.95 },**

**{ "Title-02", "Author-02", 2008, 90.25 },**

**{ "Title-03", "Author-03", 2002, 180.50 }}**

**;**

**int n = 3; // кількість елементів у масиві**

**BOOK C[3]; // інший масив, з якого буде виконуватись читання**

**int n2; // кількість елементів у масиві C**

**int i;**

**// 1. Запис масиву об’єктів в файл**

**// outF - екземпляр файлу, в який здійснюється запис**

**ofstream outF(filename, ios::out | ios::binary);**

**if (!outF) return false;**

**// записати значення n**

**outF.write((char\*)&n, sizeof(int));**

**// запис масиву B[] в файл wf**

**for (i = 0; i < n; i++)**

**{**

**outF.write((char\*)&(B[i]), sizeof(BOOK));**

**}**

**cout << "Array is written\n" << endl;**

**// після закінчення роботи з файлом його потрібно закрити (обов'язково)**

**outF.close();**

**// 2. Читання масиву структур з файлу**

**// inF - екземпляр файлу, з якого здійснюється читання**

**ifstream inF(filename, ios::in | ios::binary);**

**if (!inF) return false;**

**cout << "Read the array...\n";**

**// Спочатку прочитати кількість записаних структур**

**inF.read((char\*)&n2, sizeof(int));**

**// цикл читання масиву об’єктів в змінну C**

**for (i = 0; i < n2; i++)**

**inF.read((char\*)&(C[i]), sizeof(BOOK));**

**inF.close(); // закрити файл**

**//**  **вивід масиву C на екран**

**cout << "Array C:" << endl;**

**for (i = 0; i < n2; i++)**

**{**

**cout << "Title = " << C[i].title << ", ";**

**cout << "Author = " << C[i].author << ", ";**

**cout << "Year = " << C[i].year << ", ";**

**cout << "Price = " << C[i].price << endl;**

**}**

**}**

**int main ()**

**{system("color F0");**

**string a="book.txt";**

**char c1[50]=" ";**

**strncpy(c1,a.c\_str(),a.length()+1);**

**if (Example7(c1))**

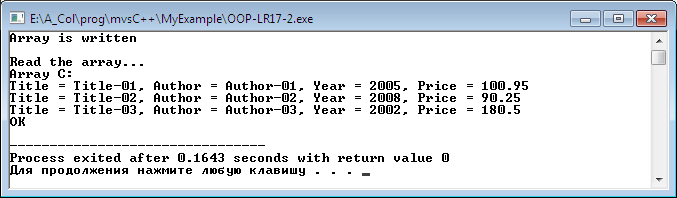
**{cout << "OK" << endl;**

**return 0;**

**}else return 1;**

**}**

Результат роботи



**Приклад 2.** Запису/читання масиву чисел типу double

Демонструється:

* запис у файл масиву **M** чисел типу double функцією write();
* читання з файлу масиву чисел типу double функцією read().

Файл відкривається у двійковому форматі.

**#include <iostream>>**

**#include <fstream>**

**using namespace std;**

**// використання функцій read(), write() для запису/читання масиву чисел**

**bool Example8(const char \* filename)**

**{**

**double M[] = { 2.44, 3.85, -3.23, 11.85, 3.38 }; // масив чисел**

**int i;**

**int n = 5; // кількість елементів у масиві M**

**// 1. Запис масиву в файл**

**// 1.1. Створити екземпляр outF, зв'язаний з файлом filename**

**ofstream outF(filename, ios::out | ios::binary); // для запису, двійковий формат**

**// 1.2. Перевірка, чи відкривається файл**

**if (!outF)**

**{**

**cout << "Error. Cannot open the file.";**

**return false;**

**}**

**// 1.3. Записати кількість елементів у масиві M**

**outF.write((char\*)&n, sizeof(int));**

**// 1.4. Записати увесь масив в файл**

**outF.write((char\*)&M, sizeof(double)\*n);**

**outF.close(); // закрити файл**

**// 2. Читання даних з файлу filename в масив M2**

**double M2[5];**

**int n2;**

**// 2.1. Відкрити файл для читання**

**ifstream inF(filename, ios::in | ios::binary);**

**// 2.2. Перевірка чи файл відкрито**

**if (!inF)**

**{**

**cout << "Error. Cannot open file.";**

**return false;**

**}**

**// 2.3. Зчитати к-сть елементів у масиві**

**inF.read((char\*)&n2, sizeof(int));**

**// 2.4. Зчитати дані з файлу в масив M2**

**inF.read((char\*)&M2, sizeof(double)\*n2);**

**inF.close(); // закрити файл**

**// 2.5. Вивести масив M2 на екран**

**cout << "Array M2:\n";**

**for (i = 0; i < n2; i++)**

**cout << M2[i] << " ";**

**cout << endl;**

**return true;**

**}**

**void main()**

**{**

**Example8("file8.bin");**

**}**

Результат роботи програми:

Array M2:

2.44 3.85 -3.23 11.85 3.38