

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра ЕОМ



Звіт
до лабораторної роботи № 2
з дисципліни: «Програмування, частина 2 (ООП)»
на тему: «Потоковий Ввід-Вивід»
Варіант № 14

Підготував: Мишак М.А.
студент групи КІ-103

Перевірив:
Ст. викладач
Каф. ЕОМ
Гузинець Н.В

Мета: познайомитися із потоковим вводом-виводом.

Теоретичний матеріал

У мові програмування C++ так само як і у багатьох об'єктно орієнтованих мовах програмування існує поняття просторів імен, що розділяються, на відміну від мови C в якій існує один глобальний простір імен. Простори імен служать для об'єднання класів, що написані різними програмістами або мають схоже призначення в групі. Завдяки цьому з'являється можливість усувати неоднозначності, що пов'язані з використанням класів, що мають однакові імена, але різне призначення. Також простори імен дозволяють безпечно підключати бібліотеки класів не боячись співпадінь у назвах існуючих і підключених класів. Все, що оголошене в межах простору імен є видимим лише в його межах.

Простори імен є відкритими. Це означає, що можна в будь-якому місці програми оголосити простір імен з існуючою назвою, додати в нього нові елементи і при компіляції ці простори імен об'єднуються в один, що міститиме існуючі і нововведені елементи

Щоб постійно не вказувати простори імен при зверненні до їх елементів, їх можна підключити до програми за допомогою ключового слова `using` частково або повністю. Для підключення окремих елементів простору імен використовується `using` оголошення. Для підключення всього простору імен з використовується `using` директива.

При підключенні всього простору імен ми можемо звичним чином звертатися до всіх його елементів. Проте, при підключенні кількох просторів імен, що мають елементи з однаковою назвою до яких відбувається звертання, відбудеться конфлікт імен. Для його усунення слід явно вказувати до якого простору імен належить елемент до якого іде звертання.

Система вводу-виводу в стандартній бібліотеці C++ реалізована у вигляді потоків. Потік вводу-виводу – це логічний пристрій, який приймає та видає інформацію користувача. Кожен потік зв'язаний з фізичним пристроєм (клавіатура, монітор) або з файлом. Бібліотека потоків `iostream` реалізована як ієрархія класів та забезпечує широкі можливості для виконання операцій вводу-виводу.

Щоб мати можливість використовувати стандартні потоки необхідно підключити заголовочний файл `iostream` або `iostream.h`. Загалом різниця між стандартними заголовочними файлами з розширенням `*.h` і без нього полягає в тому, що файли з розширенням `*.h` відносяться до мови C, а без розширення – до C++. Таким чином програмуючи на мові C++ безпечніше використовувати заголовочні файли без розширення `*.h`, які орієнтовані на мову C++. Проте в цьому випадку може бути необхідним підключати додатково простори імен. При використанні стандартних бібліотек вводу виводу таким простором імен є `std`.

Система вводу-виводу дозволяє виконувати форматування даних та змінювати визначені параметри вводу інформації. Дані операції реалізовані за допомогою функцій форматування, прапорців та маніпуляторів.

Крім вже описаних функцій, бібліотека вводу-виводу C++ містить широкий набір інших функцій.

Робота з файлами в мові C++ як і у мові C передбачає 3 етапи: відкриття файлу (файлового потоку), обмін даними з файловим потоком, закриття файлового потоку.

Завдання: З клавіатури вводиться найменування жорсткого диску, його об'єм і відсоток зайнятого простору, а у файл записується найменування жорсткого диску, його об'єм і кількість байт зайнятого простору.

Лістинги (тексти) програм:

```
#include <iostream>
#include <fstream>
#include <iomanip>
#include <string>

using namespace std;

const long long GB = 1073741824;

long long func(long long volume, double percentage) {
    return static_cast<long long>(volume * (percentage / 100.0));
}

void showData() {
    ifstream infile("lab2.txt");
    if (!infile.is_open()) {
        cout << "File isn't open or doesn't exist." << endl;
        return;
    }

    cout << left << setw(15) << "Drive Name"
         << setw(15) << "Volume (GB)"
         << setw(25) << "Occupied Space (bytes)" << endl;
    cout << string(55, '-') << endl;

    string drive_name;
    double volume;
    long long occupied;

    while (infile >> drive_name >> volume >> occupied) {
        cout << left << setw(15) << drive_name
             << setw(15) << volume
             << setw(25) << occupied << endl;
    }
    infile.close();
}

void writeData() {
    ofstream file("lab2.txt", ios_base::app);
    if (!file.is_open()) {
```

```

        cout << "File isn't open." << endl;
        return;
    }

    string drive_name;
    double volume, percentage;

    cout << "Enter drive name: ";
    cin >> drive_name;

    cout << "Enter volume (GB): ";
    cin >> volume;

    cout << "Enter percentage of occupied space: ";
    cin >> percentage;

    long long volume_bytes = static_cast<long long>(volume * GB);
    long long occupied_bytes = func(volume_bytes, percentage);

    file << drive_name << " " << volume << " " << occupied_bytes << endl;
    file.close();
}

int main() {
    while (true) {
        cout << "Show data - press 1" << endl;
        cout << "Write data - press 2" << endl;
        cout << "Exit - press 3" << endl;

        int choice;
        cout << "Enter choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
                showData();
                break;
            case 2:
                writeData();
                break;
            case 3:
                cout << "Program is closing" << endl;
                return 0;
            default:
                cout << "Invalid choice, please try again." << endl;
        }
    }
}

```

Результати виконання програм:

```
Консоль отладки Microsoft V x + v
Show data - press 1
Write data - press 2
Exit - press 3
Enter choice: 2
Enter drive name: D
Enter volume (GB): 512
Enter percentage of occupied space: 50
Show data - press 1
Write data - press 2
Exit - press 3
Enter choice: 1
Drive Name      Volume (GB)      Occupied Space (bytes)
-----
D                512              274877906944
Show data - press 1
Write data - press 2
Exit - press 3
Enter choice: 3
Program is closing

E:\VS22\Lab2c++\x64\Debug\Lab2c++.exe (процесс 50144) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

Висновок: На даній лабораторній роботі я познайомився з потоковим вводом-виводом у мові програмування C++. На основі цих знань створив програму яка записує дані в файл, після чого може читати ці дані і закривати саму програму.