

Додаткова лекція

СИСТЕМА ФАЙЛОВОГО ВВОДУ / ВИВОДУ

В **C++** введення/виведення даних в файл чи з файлу реалізується через **ПОТОКИ**

Загальна схема така:

■ Створюється потік вводу/виводу (об'єкти класів **ifstream** (потік вводу), **ofstream** (потік виводу) або **fstream** (потік вводу-виводу)).

Необхідно підключити заголовок **<fstream>**

■ З потоком необхідно зв'язати **файл**:

1) за допомогою метода **open()**

2) при створенні об'єкта **аргументом конструктора** вказують ім'я файлу

■ Після завершення роботи файл закривають за допомогою метода **close()**

ПРИКЛАД

```
#include <iostream>
#include <fstream>
using namespace std;
int main(){
char word[30]; // масив для зчитув. слів
int i=1;      // індексна змінна
//Потік виводу в файл mydata.txt:
ofstream out("D:\\mydata.txt");
//Перевірка відкриття файлу:
if(!out) {cout<<"File mydata.txt is not
opened.\n";
return 1;}
//Запис даних в файл mydata.txt:
else {out<<"This file contains simple text";
//Закриття файлу для запису:
out.close();}
```

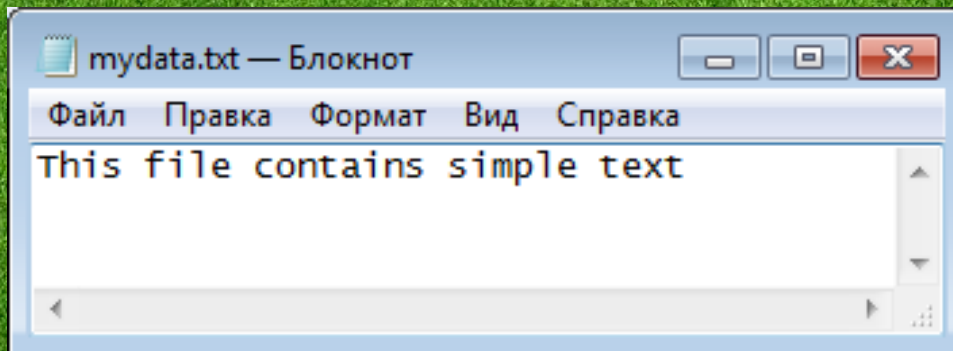
ПРИКЛАД - продовження

```
//Потік виводу в файл newdata.txt:  
ofstream outMore;  
outMore.open("D:\\newdata.txt");  
//Перевірка відкриття файлу newdata.txt:  
if(!outMore) {cout<<"File newdata.txt is not  
opened.\n";  
return 1;}  
ifstream in; // потік виводу з файлу mydata.txt:  
//Перевірка відкриття файлу mydata.txt:  
in.open("D:\\mydata.txt");  
if(!in) {cout<<"File mydata.txt is not  
opened.\n";  
return 1;}
```

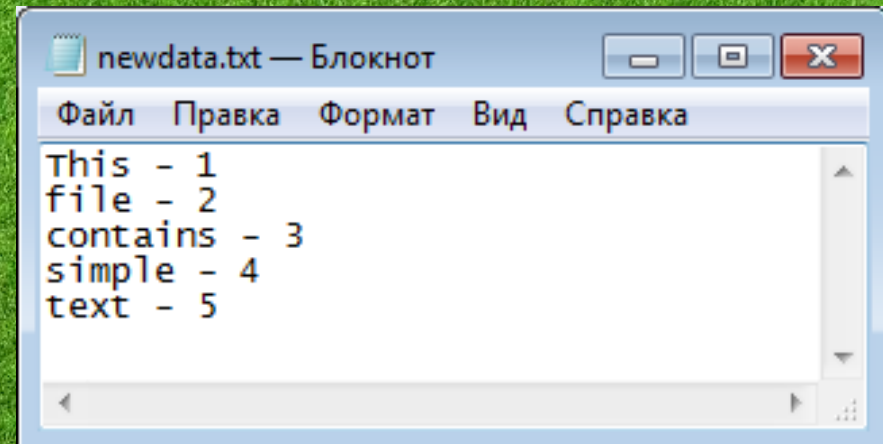
ПРИКЛАД - продовження

```
//Зчитування з mydata.txt і запис в newdata.txt:  
while(!in.eof()){  
in>>word;  
outMore<<word<<" - "<<i<<endl;  
i++;}  
//Закриття файлу mydata.txt:  
in.close();  
//Закриття файлу newdata.txt:  
outMore.close();  
return 0;}
```

Запустити на
виконання



```
mydata.txt — Блокнот  
Файл  Правка  Формат  Вид  Справка  
This file contains simple text
```



```
newdata.txt — Блокнот  
Файл  Правка  Формат  Вид  Справка  
This - 1  
file - 2  
contains - 3  
simple - 4  
text - 5
```

Пояснення

■ Оператор вводу `>>` зчитує текст (з символного масиву) до пробілу або кінця тексту.

Приклад: команда `in>>word`

■ Метод `eof()` повертає значення `true` при досягненні кінця файлу

■ Деякі корисні методи:

`put()` – запис символу в файл

`get()` – зчитування символу з файла

`read()` – зчитування з бінарного файла

`write()` – запис в бінарний файл

Режими запису/зчитування

- Існують різні режими зчитування/запису даних з файлів та в файли
- Режим визначається другим необов'язковим аргументом методу `open()`
- Значення цього параметра наведені далі в таблиці. Їх можна комбінувати за допомогою оператора **логічного або** (тобто `|`)
- Якщо файл відкривається через потік на основі об'єкта класу `ofstream` вміст файлу знищується (видаляється)

Значення параметра **способу відкриття файлу**

Значення	Пояснення
ios::app	В цьому режимі дані дописуються в кінець файлу
ios::ate	Режим, в якому відбувається пошук кінця файлу, а введення може відбуватись в любе місце файлу
ios::binary	Файл відкривається в бінарному режимі
ios::in	Режим відкриття файлу для вводу
ios::out	Режим відкриття файлу для виводу
ios::trunc	Режим с сигналізацією про те, що попередній вміст файлу буде знищено