Tới bây giờ, chúng ta đã sử dụng thư viện chuẩn **iostream**, cung cấp các phương thức **cin** và **cout** để đọc từ Standard Input và ghi tới Standard Output tương ứng.

Chương này sẽ hướng dẫn bạn cách đọc và ghi một file. Điều này cần một Thư viện chuẩn C++ khác là **fstream**, mà định nghĩa 3 kiểu dữ liệu mới:

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Miêu tả** |
| ofstream | Kiểu dữ liệu này biểu diễn Output File Stream và được sử dụng để tạo các file và để ghi thông tin tới các file đó |
| ifstream | Kiểu dữ liệu này biểu diễn Input File Stream và được sử dụng để đọc thông tin từ các file |
| fstream | Kiểu dữ liệu này nói chung biểu diễn File Stream, và có các khả năng của cả ofstream và ifstream, nghĩa là nó có thể tạo file, ghi thông tin tới file và đọc thông tin từ file |

Để thực hiện tiến trình xử lý file trong C++, bạn bao các header file là <iostream> và <fstream> trong source file của chương trình C++ của bạn.

Mở một File trong C++

Một file phải được mở trước khi bạn có thể đọc thông tin từ nó hoặc ghi thông tin tới nó. Hoặc đối tượng **ofstream** hoặc đối tượng **fstream** có thể được sử dụng để mở một file với mục đích viết hoặc đối tượng ifstream được sử dụng để mở file chỉ với mục đích đọc.

Dưới đây là cú pháp chuẩn cho hàm open(), là một thành viên của các đối tượng fstream, ifstream và ofstream trong C++:

void open(const char \*ten\_file, ios::che\_do);

Tại đây, tham số đầu tiên xác định tên và vị trí của file để được mở và tham số thứ hai của hàm thành viên **open()** định nghĩa chế độ mà file nên được mở.

|  |  |
| --- | --- |
| **Chế độ** | **Miêu tả** |
| ios::app | Chế độ Append. Tất cả output tới file đó được phụ thêm vào cuối file đó |
| ios::ate | Mở một file cho outpur và di chuyển điều khiển read/write tới cuối của file |
| ios::in | Mở một file để đọc |
| ios::out | Mở một file để ghi |
| ios::trunc | Nếu file này đã tồn tại, nội dung của nó sẽ được cắt (truncate) trước khi mở file |

Bạn có thể kết hợp hai hoặc nhiều giá trị này bằng việc hoặc chúng cùng với nhau (sử dụng (|). Ví dụ, nếu bạn muốn mở một file trong chế độ ghi và muốn cắt (truncate) nó trong trường hợp nó đã tồn tại, bạn theo cú pháp sau:

ofstream outfile;

outfile.open("file.dat", ios::out | ios::trunc );

Theo cách tương tự, bạn có thể mở một file với mục đích đọc và ghi như sau:

fstream vietjack;

vietjack.open("file.dat", ios::out | ios::in );

Đóng một File trong C++

Khi một chương trình C++ kết thúc, nó tự động đóng tất cả Stream, giải phóng tất cả bộ nhớ đã cấp phát và đóng tất cả file đã mở. Nhưng nó là bài thực hành tốt cho một lập trình viên nên đóng tất cả file đã mở trước khi kết thúc chương trình.

Dưới đây là cú pháp chung cho hàm **close()** trong C++, là một thành viên của các đối tượng fstream, ifstream và ofstream trong C++:

void close();

Ghi File trong C++

Trong khi lập trình C++, bạn ghi thông tin tới một file từ chương trình của bạn bởi sử dụng toán tử chèn luồng là (<<), giống như khi bạn sử dụng toán tử đó để tạo thông tin đầu ra tới màn hình. Chỉ có một điểm khác nhau là bạn sử dụng một đối tượng **ofstream** hoặc **fstream** trong C++ thay cho đối tượng **cout** trong C++.

Đọc một File trong C++

Bạn đọc thông tin từ một file trong chương trình C++ của bạn bởi sử dụng toán tử trích luồng là (>>), giống như bạn sử dụng toán tử đó để nhập thông tin đầu vào từ bàn phím. Điểm khác nhau là bạn sử dụng một đối tượng **ifstream** hoặc **fstream** thay vì sử dụng đối tượng **cin** trong C++.

Ví dụ về Đọc và Ghi file trong C++

Chương trình C++ sau mở một file trong chế độ đọc và ghi. Sau khi ghi thông tin được nhập vào bởi người sử dụng tới một file là vietjack.dat, chương trình đọc thông tin từ file đó và tạo kết quả đầu ra trên màn hình:

#include <fstream>

#include <iostream>

using namespace std;

int main ()

{

char data[100];

// mo mot file trong che do write.

ofstream outfile;

outfile.open("vietjack.dat");

cout << "Ghi du lieu toi file!" << endl;

cout << "Nhap ten cua ban: ";

cin.getline(data, 100);

// ghi du lieu da nhap vao trong file.

outfile << data << endl;

cout << "Nhap tuoi cua ban: ";

cin >> data;

cin.ignore();

// ghi du lieu da nhap vao trong file.

outfile << data << endl;

// dong file da mo.

outfile.close();

// mo mot file trong che do read.

ifstream infile;

infile.open("vietjack.dat");

cout << "\n===========================\n" ;

cout << "Doc du lieu co trong file!" << endl;

infile >> data;

// ghi du lieu tren man hinh.

cout << data << endl;

// tiep tuc doc va hien thi du lieu.

infile >> data;

cout << data << endl;

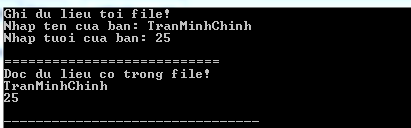
// dong file da mo.

infile.close();

return 0;

}

Biên dịch và chạy chương trình C++ trên sẽ cho kết quả sau:



Ví dụ trên sử dụng các hàm bổ sung từ đối tượng cin, như hàm getline() để đọc dòng từ bên ngoài và hàm ignore() để bỏ qua các ký tự phụ bên trái lệnh read trước đó.

Con trỏ vị trí File trong C++

Cả hai đối tượng **istream** và **ostream** đều cung cấp các hàm thành viên để xác định lại vị trí của con trỏ vị trí file (file-position pointer). Các hàm thành viên này là **seekg** (viết tắt của seek get) cho istream và **seekp** (viết tắt của seek put) cho ostream trong C++.

Tham số cho seekg và seekp thương là một long int. Tham số thứ hai có thể được xác định để chỉ dẫn hướng tìm kiếm. Hướng tìm kiếm có thể là **ios::beg** (mặc định) để xác định vị trí liên quan tới phần bắt đầu của một Stream, là **ios::cur** để xác định vị trí liên quan tới vị trí hiện tại trong một Stream hoặc là **ios::end** để xác định vị trí liên quan tới phần kết thúc của một Stream trong C++.

Con trỏ vị trí file là một giá trị integer mà xác định vị trí trong file, tính toán là số byte từ vị trí bắt đầu của file đó. Dưới đây là một số ví dụ để xác định vị trí của con trỏ vị trí file trong C++: