# KIỂU TỆP (FILE)

1. **Khái niệm:** Tệp là tập hợp các dữ liệu được tổ chức theo một cách thức nhất định và được lưu trữ trên bộ nhớ ngoài. Có hai dạng tệp: tệp văn bản và tệp có cấu trúc. Trong chương trình ta chỉ đề cập đến tệp văn bản (TEXT)
2. **Tệp văn bản:** Là tập hợp các kí tự thuộc bảng mã **ASCII** được tổ chức thành từng dòng. Kí tự kết thúc dòng là **‘\n’** . Việc đọc/ ghi dữ liệu trên tệp văn bản thường theo từng dòng.

#### Luồng dữ liệu:

Trong **C++** ta đã biết đến cách nhập/ xuất dữ liệu bằng lệnh **cin** , **cout**. Các lệnh này giao tiếp với chương trình thông qua bàn phím và màn hình.

Để nhập/ xuất dữ liệu thông qua tệp văn bản ta cần sử dụng các luồng vào/ ra được xây dựng trong **header fstream**. Khi đó ta có các luồng sau đây:

* + **ifstream**: Luồng vào, để đọc dữ liệu từ tệp.
  + **ofstream**: Luồng ra, để ghi dữ liệu lên tệp.

#### Ví dụ về cách đọc, ghi dữ liệu bằng tệp

**Ví dụ 1:** Giả sử ta tạo sẵn một tệp **input.txt** trên đĩa D có cấu trúc như sau:

|  |  |
| --- | --- |
| **Cấu trúc** | **Ví dụ** |
| Dòng 1 là một chuỗi kí tự | **Khanh** |
| Dòng 2 là một số nguyên n không quá 100 | **6** |
| Dòng 3 là n số thực a1 , a2 , a3 …, an Dòng 4 là một kí tự | **3.2 6.1 5.3 7.5 8.8**  **T** |

**Yêu cầu:** Hãy tạo một chuỗi S để lưu dòng 1, một số nguyên n để lưu trữ dòng 2, một mảng để lưu dòng 3 và một biến kiểu char để lưu dòng 4.

Ghi lên tệp “**output**.**txt**” một số là trung bình cộng của **n** số **a1 , a2 , a3 …, an**

#### Bước 1: Khai báo

**#include <iostream>**

**#include <fstream> //Khai báo header fstream để sử dụng nhâp/xuất trên file using namespace std;**

**string s; //Biến s kiểu string để chứa chuỗi ở dòng 1 int n;**

**float a[101]; //Mảng a chứa được 101 phần tử. Ta khai báo 101 vì char c; mảng a bắt đầu từ a[0]**

**Bước 2: Đọc dữ liệu từ file** input.txt **ifstream fi("input.txt"); fi>>s;**

**fi>>n;**

**for (int i=1;i<=n;++i) fi>>a[i]; fi>>c;**

Như vậy thay vì nhập từ luồng chuẩn(từ bàn phím) với lệnh **cin** thì ta chỉ cần đổi hướng sang luồng **fi** (là một luồng kiểu file văn bản). Đến đây ta có thể xuất dữ liệu ra màn hình theo luồng chuẩn với lệnh **cout** hoặc xuất ra file văn bản theo luồng file văn bản.

#### Bước 3: Xuất dữ liệu

|  |  |
| --- | --- |
| **Xuất ra màn hình** | **Xuất ra file “**output**.**txt |
| cout <<s<<'\n';  for (int i=1;i<=n;++i)cout<<a[i]<<" "; | //Khai báo luồng ra là file output.txt fstream fo("output.out");  fo <<s<<'\n';  for (int i=1;i<=n;++i) fo<<a[i]<<" "; |
| **Kết quả:**  **Khanh**  **3.2 6.1 5.3 7.5 8.8** | **Kết quả:** File **output**.**txt Khanh**  **3.2 6.1 5.3 7.5 8.8** |

Một cách đơn giản hơn là ta định hướng lại luồng chuẩn thành luồng file bởi lệnh **freopen()** . Khi đó các lệnh **cin, cout** được chuyển hướng trở thành các lệnh nhập/ xuất ra file văn bản chứ không theo luồng chuẩn nữa.

int main()

{

freopen("input.txt","r",stdin); //Thay luồng nhập chuẩn thành luồng file freopen("output.txt","w",stdout); //Thay luồng xuất chuẩn thành luồng file cin>>s;

cin>>n;

for (int i=1;i<=n;++i) cin>>a[i]; cin>>c;

// cout << s << '\n';

for (int i=1;i<=n;++i) cout<<a[i]<<" "; return 0;

}

##### Ví dụ 2: Đọc tệp văn bản theo từng dữ liệu rời rạc liên tiếp.

Giả sử ta tạo sẵn một tệp input.txt trên đĩa D có cấu trúc như sau:

|  |  |
| --- | --- |
| **Cấu trúc** | **Ví dụ** |
| Dòng 1 ghi 2 số nguyên kiểu int  Dòng 2 ghi hai số nguyên kiểu long Dòng 3 ghi 3 số thực kiểu double | 1 5  3291 -4327  -3.4 2.1 8.98 |

**Yêu cầu:** Hãy tạo 2 số nguyên kiểu int để lưu trữ dòng 1, hai biến kiểu long để lưu dòng 2 và 3 biến double để lưu dòng 3. Ghi lên file “output.txt” tổng của 4 số nguyên ở dòng thứ nhất và tổng của hai số thực lên dòng thứ 2.

#### Một số thao tác đặc biệt với kiểu tệp

Lệnh **ofstream fo(“tên tệp”, ios::app):** Mở tệp ở chế độ ghi nối dữ liệu vào cuối tệp

Lệnh **ofstream fo(“tên tệp”, ios::noreplace):** Mở tệp ở chế độ bảo vệ. Nếu tệp đã có thì tệp cũ không bị xóa. Chế độ mở kiểu này thường đi kèm với việc sử dụng **(!fo)**

#### Ví dụ

**ofstream fo(“D:\\input.txt”, ios::noreplace); if (!fo)**

**{**

**cout<<”Tep da ton tai, khong the ghi de len duoc”; return;**

**}**

Lệnh **ofstream fo(“tên tệp”, ios::nocreate):** Kiểm tra tệp đã tồn tại chưa. Nếu chưa tồn tại, biến fo sẽ nhận giá trị false và tệp đã sẵn sàng để ghi dữ liệu được.

#### Ví dụ

**ofstream fo(“D:\\input.txt”, ios::nocreate); if (!fo)**

**{**

**cout<<”Tep chua ton tai”; //Có thể ghi dữ liệu vào tệp D:\\input.txt”.**

**…**

**}**

#### Truy cập ngẫu nhiên

Giả sử có biến tệp f. Ta có một số phương thức thường dùng:

|  |  |  |
| --- | --- | --- |
| Truy cập vào đầu tệp | f.seekg(0) | cin.seekg(0) |
| Truy cập tới vị trí pos | f.seekg(pos) | cin.seekg(pos) |
| Kiểm tra kết thúc tệp | eof(f) | cin.eof |