Tài liệu tham khảo :

**1. Ghi file, đọc file:**

- **Trong C++, để đọc và lưu file, chúng ta sử dụng các đối tượng file của thư viện fstream hay ifstream và ofstream, cụ thể như sau:**

+) Thư viện fstream trong C++ là một trong các thư viện chuẩn (Standard Library) của ngôn ngữ lập trình C++. Thư viện này cung cấp các thuật toán và hàm để thao tác với các file, như đọc và ghi dữ liệu vào file.

+) ifstream và ofstream là hai đối tượng trong thư viện fstream. ifstream (input file stream) được sử dụng để đọc dữ liệu từ file, trong khi ofstream (output file stream) được sử dụng để ghi dữ liệu vào file.

* Trong C++ , các cú pháp thường được sử dụng như sau:

+) Để mở file:

* ofstream outfile; // Khai báo thư viện, tạo ra một đối tượng  ofstream có tên là outfile và mở file "filename.txt".
* outfile.open("filename.txt"); // Để tạo một tập tin mới
* outfile.open("filename.txt", ios::app); // Để thêm vào một tập tin hiện có, ghi thêm nội dung vào cuối file đó thay vì ghi đè lên nội dung hiện có trong file.
* ifstream infile; //Khai báo thư viện, tạo một đối tượng ifstream có tên là infile và mở file "filename.txt".
* infile.open("filename.txt"); // cho phép mở file "filename.txt" để đọc các dữ liệu bên trong.

+) Để đọc các ký tự từ file:

* char ch; while(infile.get(ch)) { cout << ch; } // Đầu tiên, đọc ký tự đầu tiên bằng hàm get() của đối tượng ifstream. Hàm get() trả về ký tự đọc được và đẩy đầu đọc file một ký tự. Sau đó, in ký tự này ra màn hình bằng cout. Tiếp theo, vòng lặp tiếp tục đọc ký tự tiếp theo từ file, in nó ra màn hình và tiếp tục như vậy cho đến khi đọc hết toàn bộ file.

+) Để ghi dữ liệu vào file:

* outfile << "Hello world!";

+) Để đóng file:

* infile.close();
* outfile.close();

- **Khi sử dụng đọc file và ghi file trong chương trình, cần chú ý các điểm sau:**

+) Kiểm tra đường dẫn tệp tin:

* Đảm bảo rằng tệp tin cần đọc hoặc viết được chỉ định chính xác và ở đúng vị trí.

+) Kiểm tra đường dẫn thư mục:

* Đảm bảo rằng người dùng đang chạy chương trình của bạn có đủ quyền truy cập vào thư mục, trong đó tệp tin cần đọc hoặc viết được lưu trữ.

+) Mở tệp tin:

* Sử dụng đối tượng ifstream (đọc) hoặc ofstream (ghi) để mở tệp tin.
* Kiểm tra xem việc mở tệp tin có thành công hay không.
* Cho phép chỉ đọc hoặc ghi dữ liệu.

+) Đọc tệp tin:

* Sử dụng hàm đọc, như hàm get() hoặc getline(), để đọc dữ liệu từ tệp tin.
* Kiểm tra xem việc đọc dữ liệu có thành công hay không.
* Sử dụng vòng lặp để đọc dữ liệu từng dòng, từng ký tự hoặc theo cách khác tùy thuộc vào yêu cầu.

+) Lưu dữ liệu vào tệp tin:

* Sử dụng hàm ghi, như hàm put() hoặc endl(), để ghi dữ liệu vào tệp tin.
* Kiểm tra xem việc ghi dữ liệu có thành công hay không.

+) Đóng tệp tin:

* Đảm bảo rằng bạn đã đóng tệp tin bằng cách sử dụng hàm close().
* Hãy chắc chắn rằng bạn đã đóng tệp tin, trước khi thực hiện các thao tác khác.

+) Xử lý ngoại lệ:

* Sử dụng các hàm xử lý ngoại lệ để giải quyết các vấn đề liên quan đến phát hiện, xử lý và thông báo lỗi cho người dùng.

**2. Thư viện string trong C++ :**

**- Các hàm thường dùng trong thư viện string :**

* Lớp string: là kiểu dữ liệu chuỗi dữ liệu, được định nghĩa trong thư viện string. Lớp string cung cấp các hàm để thêm, loại bỏ ký tự, so sánh chuỗi và truy cập các phần tử trong chuỗi.
* Hàm length(), size(): được sử dụng để trả về độ dài của chuỗi ký tự trong đối tượng string.
* Hàm insert(): được sử dụng để chèn một chuỗi con vào trong đối tượng string ở một vị trí cụ thể.
* Hàm push\_back(): được sử dụng để thêm một ký tự vào cuối đối tượng string.
* Hàm to\_string() : hỗ trợ chuyển đổi các kiểu dữ liệu số sang kiểu dữ liệu chuỗi (string). Hàm to\_string() nhận một giá trị số làm đối số và trả về một đối tượng string tương ứng với giá trị số đó.

**- Những lưu ý khi sử dụng thư viện string :**

* Không sử dụng con trỏ đến mảng ký tự để thao tác với string. Khi sử dụng con trỏ đến mảng ký tự, không kiểm soát được vùng nhớ và có thể dẫn đến tràn bộ nhớ hoặc lỗi khác. Thay vào đó, sử dụng các phương thức của lớp string để thao tác với chuỗi.
* Thao tác chuỗi string với nhiều phép gán, nối chuỗi hay xóa chuỗi sẽ tốn nhiều bộ nhớ đệm và thời gian xử lý. Vì thế, khi cần thay đổi nhiều nội dung của chuỗi, nên sử dụng lớp stringstream (thuộc thư viện <sstream>) để giảm thiểu việc sao chép và gán chuỗi.
* Hạn chế sử dụng lớp string khi cần tối ưu hiệu suất chương trình, đặc biệt đối với ứng dụng trên các thiết bị nhúng có tài nguyên hạn chế. Trong trường hợp này, có thể sử dụng các cấu trúc dữ liệu khác hoặc số học bit.