**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN – ĐHQG\_HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

|  |
| --- |
|  |
|  | A blue and white logo  Description automatically generated |

**An toàn và phục hồi dữ liệu**

**Đồ án cuối kỳ**

**Giảng viên: Thái Hùng Văn**

[**Ngô Đình Hy**](https://courses.fit.hcmus.edu.vn/user/view.php?id=15630&course=1)

**MSSV** **Họ và tên**

20120083 Nguyễn Trọng Hiếu

20120144 Lê Chí Nghĩa

**Mục Lục**

[1. Tiếp cận bài toán 3](#_Toc154689715)

[2. Cấu trúc của file 3](#_Toc154689716)

[3. Các yêu cầu phụ 4](#_Toc154689717)

[4. Phương thức thiết kế 5](#_Toc154689718)

[4.1 global.h 7](#_Toc154689719)

[4.2 fileTemplate.h 9](#_Toc154689720)

[4.3 fileCombine.h 10](#_Toc154689721)

[5. Nhận xét 13](#_Toc154689722)

## Tiếp cận bài toán

Thiết kế 1 định dạng file mới (file kết hợp) với cấu trúc đặc biệt để lưu trữ dữ liệu của 2 file con một cách an toàn với các yêu cầu phụ đi kèm.

## Cấu trúc của file

***Đặt vấn đề***

* ***Vấn dề 1***: Dưới góc nhìn của một hệ thống lưu trữ tổng quát thì các kích thước của tập tin được lưu trữ phải có kích thước cố định. Tuy nhiên hai tập tin đưa vào có kích thước giãn nở được.
* ***Vấn đề 2***: Thỏa mãn được các yêu cầu phụ được nêu bên dưới.

***Cấu trúc đề xuất***

**Cấu trúc chính**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| META DATA | | | FAT | | DATA |
| COMBINE FILE | FILE  1 | FILE  2 | FILE  1 | FILE  2 |  |

**Thiết kế phân vùng DATA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | | | | |
| CLUSTER 1  *(8 SECTORS)* | CLUSTER 2  *(8 SECTORS)* | … | … | CLUSTER N  *(8 SECTORS)* |
| FILE 1 | FILE 2  *(7 SECTORS) | (1 SECTORS)* | … | … | … | … |

***Khả năng của cấu trúc***

* ***Giải quyết các vấn đề được đặt ra***
* ***Vấn đề 1:*** Với cấu trúc đề xuất, các thông tin được lưu trữ sẽ thông qua một text file có khả năng thay đổi kích thước tùy ý. Sau khi thông tin đã được lưu vào text file thì sẽ gọi cơ chế quản lý của cấu trúc lưu trữ text file đó vô trong cấu trúc. Cụ thể sẽ được nêu trong phần phương thức thiết kế.
* ***Vấn đề 2:*** Đáp ứng các yêu cầu phụ theo các tiêu chí sau
  + ***Tránh việc dồn dữ liệu:*** Dữ liệu sẽ vẫn có thể tiếp tục thêm vào khi không gian lưu trữ tương ứng của file đó của file kết hợp còn trống. Do đó không có khả năng bị dồn dữ liệu giữa 2 file.
  + ***Các phần tử đã xóa không xóa hẳn để có khả năng khôi phục lại***: Các phần tử đã xóa không hẳn đã xóa mà chỉ gán các giá trị trên cluster lưu trữ giữ liệu trong bảng fat của file tương ứng về trạng tái không còn được sử. Khi đó với việc lưu trữ mở sẽ tìm kiếm các cluster với giá trị đại diện cho cluster trống (chưa từng bị xóa) để tiến hành lưu trữ mới. Khả năng phục hồi lại sẽ phục thuộc vào thiết kế bổ sung trên phần metadata tương ứng của file trong cấu trúc chính.
  + ***Có thêm thông tin quản lý thiết lập***: Thông tin quản lý thiết lập có thể bổ sung trong phần metadata tương ứng với tập tin kết hợp cũng như từng tập tin con bên trong.
  + ***Mỗi cá nhân có tổ chức minh họa thông tin cơ bản***: Mỗi cá nhân được thiết kế với các thông tin gắn liền với nhau chẳng hạn như Struct myStruct. Thông tin đại diện cho cá nhân sẽ được nối tuần tự các giá trị bên trong struct và tiến hành mã hóa sau đó đẩy vô file tạm. Mỗi thông tin của 1 cá nhân chiếm 1 dòng trong file tạm.
  + ***Có cơ chế kiểm tra mật khẩu***: Mật khẩu được hỗ trợ cho cả file kết hợp cũng như từng file con bên trong. Hiện tại mật khẩu đang được lưu trữ dưới dạng cơ chế HashMD5 của password người dùng đưa vào.
* ***Khả năng mở rộng***
* Tùy thuộc vào mục đích người dùng mà có thể thêm các thông tin quản lý tập tin con vào các phần metadata tương ứng.
* Đối với trường hợp không xóa hẳn giữ liệu, thiết kế được đưa ra nhằm mục đích giải quyết vấn đề này, tuy nhiên do thời hạn thực hiện đồ án nên vẫn chưa thực hiện được mục đích trên.

## Các yêu cầu phụ

* Tránh việc dồn dữ liệu
* Các phần tử được xóa sẽ không xóa hẳn để có khả năng phục hồi lại (ngoại trừ tình huống đặc biệt cần phải xóa hẳn, các phần tử đã xóa quá lâu cũng không cần phải phục hồi).
* Mỗi cá nhân có thể tổ chức minh họa vài thông tin cơ bản (Mã, Họ Tên, Ngày sinh, Ngày tham gia, Số ĐT, Số CCCD, …), trong đó số CCCD và số ĐT cần bảo mật.
* Nên có thêm các thông tin quản lý cần thiết, như ngày tạo lập, thời điểm cập nhật, công thức /key mã hóa /giải mã, …
* Cần có cơ chế kiểm tra mật khẩu động hoặc passkey mỗi khi file được mở, có khống chế thời gian (nhập sai nhiều lần thì phải đợi một thời gian sau mới có thể nhập tiếp, vẫn sai nữa thì thời gian đợi bị tăng thêm).

## Phương thức thiết kế

Được cụ thể hóa thông qua các file sau:

--------------------------

final\_lab

|-global.h // định nghĩa về các thông tin dùng cho toàn bộ các file

|-fileTemplate.h // tạo định dạng mẫu lưu trữ cho 1 file con

|-fileCombine.h // tạo định dạng lưu trữ cho file combine

|-main.cpp // chương trình demo final\_lab

--------------------------

Kết quả demo của hàm main:

A screenshot of a computer

Description automatically generated

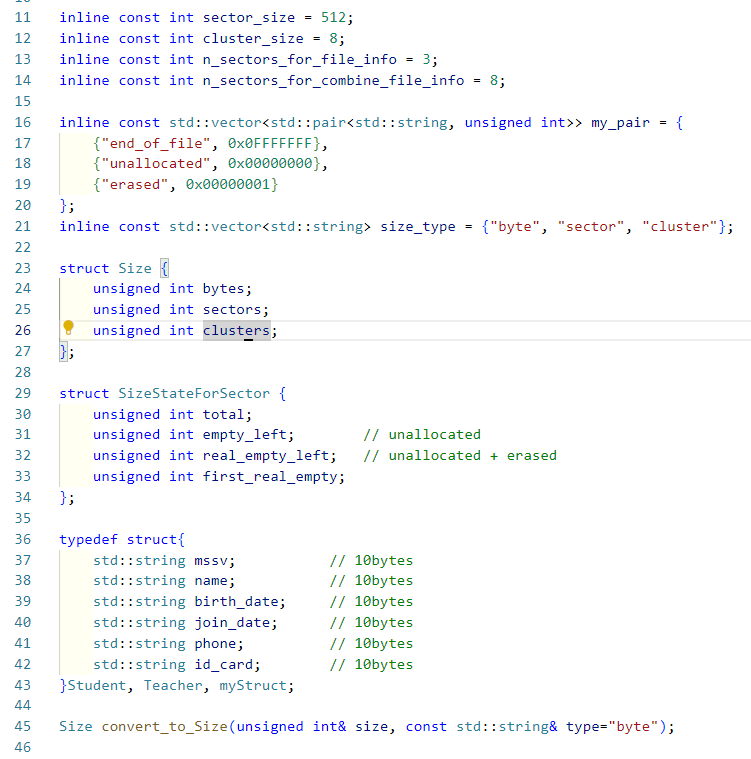
Hình : Chạy thử hàm main, một số thông tin liên quan được thể hiện trong phần [kỹ thuật làm việc với hệ thống](#ktlvvht)

A screenshot of a computer screen

Description automatically generated

Hình : File tổng hợp được thể hiện dưới dạng nhị phân

### global.h



Hình : Tổng quát của global.h

A screenshot of a computer code

Description automatically generated

Hình : Khởi tạo một đối tượng bất kì

### fileTemplate.h

A screenshot of a computer code

Description automatically generated

Hình : Tổng quát của fileTemplate.h

Trong class template này được định nghĩa biến trạng thái state, khi state = true tương ứng với file sẽ chiếm kích thước lớn và thường xuyên được đọc ghi, ngược lại (state = false) thì file sẽ chiếm kích thước nhở và ít được truy suất hơn.

Các sector của metadata cho từng file còn trống tương đối nhiều, tùy thuộc vào mục đích người dùng có thể tiếp tục tận dụng để lưu trữ các thông tin liên quan.

### fileCombine.h

A screenshot of a computer

Description automatically generated

Hình : Tổng quát của fileCombine.h

***Kỹ thuật làm việc với hệ thống***

Coi như hệ thống file combine như 1 chương trình quản lý và có pwd riêng. VD khi so sánh với hệ diều hành, mật khẩu để login vô user sử dụng tương ứng với mật khẩu để khởi tạo cũng như sử dụng của file kết hợp, mật khẩu của file con giống như mật khẩu của các tập tin lưu trữ bên trên màn hình máy tính.

1. **Thêm – xóa – sửa**

Đối với các lệnh thao tác với dữ liệu (thêm/xóa/sửa): thông qua đọc ghi bằng cluster – sector để thao tác. Do đó cần thiết lập cơ chế đọc cluster theo cấu trúc data được thiết lập bên trên (7 sectors đầu của cluster dành cho file 1 và 1 sector cuối của cluster dành cho file 2)

A screenshot of a computer program

Description automatically generated

Có các đề xuất sau cho hướng tiếp cận trong việc thao tác với thông tin:

* Dựa vào thông tin đầu vào, tiến hành truy suất đến cluster tương ứng và chỉ lấy nội dung của cluster đó ra và chỉnh sửa. Yêu cầu thông tin đầu vào được qua các quá trình mã hóa và kiểm tra kỹ lưỡng. VD: chỉnh sửa thông tin của sv X thì phải biết trước đó sv X thì phải biết lần gần nhất sv X đã được lưu ở byte thứ bao nhiêu trong file 1. (quá trình thực hiện ko tiếp cận theo hướng này).
* Đọc hết toàn bộ thông tin tương ứng cần chỉnh sửa: truy xuất đến vị trí cần chỉnh sửa, tiến hành sửa và lưu lại. (chương trình demo được thực hiện theo hướng tiếp cận này).
* Việc thực hiện xóa/sửa được thiết kế tương tự như việc thêm file như bên dưới.

A computer screen shot of a program code

Description automatically generated

Hình : Thêm đối tượng vào file tương ứng

1. **Liệt kê một đoạn trong danh sách**

* Thực hiện đọc toàn bộ file với tên và pwd tương ứng của file ra file tạm. Tùy vào mục đích cá nhân mà có thể tiến hành lọc, truy suất từng cá nhân tương ứng và in ra màn hình.

**A computer code with text

Description automatically generated with medium confidence**

1. **Đổi mật khẩu**

* Do yêu cầu không rõ ràng nên chỉ demo việc đổi mật khẩu cho file 1-2 (bên trong file kết hợp). Hoàn toàn có thể đổi mật khẩu cho file kết hợp theo phương pháp tương tự. (chỉ cần trỏ đúng đến vị trí lưu trữ mật khẩu của file tương ứng) và lưu ý một số chỗ liên quan đến mật khẩu cũ.

**A computer screen shot of a code

Description automatically generated**

1. **Đổi Cơ chế mã hóa – hiện tại không hỗ trợ (do bị hạn chế về mặt thời gian).**

Có các đề xuất sau cho việc sử dụng chơ chế mã hóa:

* Mã hóa từng đối tượng trước khi đưa vào file tạm sau đó mới tiến hành lưu file tạm vào file kết hợp.
* Khi có được file tạm lưu trữ các thông tin người dùng rồi, tiến hành mã hóa toàn bộ file tạm rồi mới lưu vào file kết hợp.

## Nhận xét

Về mặt thiết kế của hệ thống đáp ứng mọi yêu cầu phụ cũng như yêu cầu chính.

Về mặt thực thi đáp ứng được các nhu cầu cơ bản cần thiết (thay đổi nội dung như thêm đối tượng, thay đổi mật khẩu, có cơ chế kiểm tra an toàn với mật khẩu, thao tác được với từng vùng nhớ và có thể kiểm soát từng vị trí cụ thể thông quá các hàm của sector và cluster hoặc tham chiếu trực tiếp đến đựa chỉ được định nghĩa, ...) với khả năng mở rộng và tính ứng dụng cao.