Mục lục

[1 Thông tin nhóm 2](#_Toc157804405)

[2 Bảng phân công công việc 2](#_Toc157804406)

[3 Đánh giá mức độ hoàn thành 3](#_Toc157804407)

[4 Mô tả các bước thực hiện 3](#_Toc157804408)

[1. Cài đặt chức năng đọc thông tin phân vùng 3](#_Toc157804409)

[2. Cài đặt chức năng hiển thị thông tin cây thư mục của phân vùng 3](#_Toc157804410)

[3. Cài đặt giao diện chính 3](#_Toc157804411)

[5 Hình ảnh demo chương trình 3](#_Toc157804412)

[6 Nguồn tham khảo 3](#_Toc157804413)

# Thông tin nhóm

**Nhóm: 11Cent**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và Tên | MSSV | Email |
| 1 | Võ Minh Khôi | 22127213 | [Vmkhoi22@clc.fitus.edu.vn](mailto:Vmkhoi22@clc.fitus.edu.vn) |
| 2 | Nhâm Đức Huy | 22127158 | [Ndhuy22@clc.fitus.edu.vn](mailto:Ndhuy22@clc.fitus.edu.vn) |
| 3 | Phan Hải Minh | 22127283 | [Phminh22@clc.fitus.edu.vn](mailto:Phminh22@clc.fitus.edu.vn) |
| 4 | Nguyễn Tấn Hoàng | 22127129 | [Nthoang22@clc.fitus.edu.vn](mailto:Nthoang22@clc.fitus.edu.vn) |
| 5 | Lê Nguyễn Minh Châu | 22127042 | [Lnmchau22@clc.fitus.edu.vn](mailto:Lnmchau22@clc.fitus.edu.vn) |

# Bảng phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Công việc | Mô tả | Người thực hiện |
| 1 | Viết báo cáo | Thiết kế bố cục, viết, tổng hợp nội dung và chỉnh sửa, đảm bảo chất lượng báo cáo | Võ Minh Khôi |
| 2 | Tổng hợp và kiểm tra | Tổng hợp và kiểm tra tính chính xác của nội dung, chỉnh sửa bố cục báo cáo | Phan Hải Minh |
| 3 | Thiết kế giao diện | Thiết kế và cài đặt giao diện console của ứng dụng | Nhâm Đức Huy |
| 4 | FAT32 – Đọc thông tin phân vùng | Cài đặt chức năng đọc thông tin phân vùng FAT32 | Nguyễn Tấn Hoàng |
| 5 | FAT32 – Hiển thị thông tin cây thư mục của phân vùng | Cài đặt chức năng hiển thị thông tin cây thư mục của phân vùng FAT32 | Nguyễn Tấn Hoàng |
| 6 | FAT32 – Hiển thị nội dung tập tin .txt và thông báo dùng phần mềm tương thích | Cài đặt chức năng hiển thị nội dung tập tin có phần mở rộng .txt và thông báo dùng phần mềm tương thích để đọc nội dung | Nguyễn Tấn Hoàng |
| 7 | NTFS – Đọc thông tin phân vùng | Cài đặt chức năng đọc thông tin phân vùng NTFS | Lê Nguyễn Minh Châu |
| 8 | NTFS – Hiển thị thông tin cây thư mục của phân vùng | Cài đặt chức năng hiển thị thông tin cây thư mục của phân vùng NTFS | Lê Nguyễn Minh Châu |
| 9 | NTFS – Hiển thị nội dung tập tin .txt và thông báo dùng phần mềm tương thích | Cài đặt chức năng hiển thị nội dung tập tin có phần mở rộng .txt và thông báo dùng phần mềm tương thích để đọc nội dung | Lê Nguyễn Minh Châu |

# Đánh giá mức độ hoàn thành

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Yêu cầu | | Mức độ hoàn thành |
| 1 | Đọc thông tin chi tiết của một phân vùng | FAT32 |  |
| NTFS |  |
| 2 | Chương trình hiển thị cây thư mục gốc gồm tên tập tin / thư mục, trạng thái, kích thước (nếu có), chỉ số sector lưu trữ trên đĩa cứng | FAT32 |  |
| NTFS |  |
| 3 | Hiển thị nội dung tập tin đối với tập tin có phần mở rộng là .txt | FAT32 |  |
| NTFS |  |
| 4 | Hiển thị thông báo dùng phần mềm tương thích để đọc nội dung đối với tập tin có phần mở rộng khác .txt | FAT32 |  |
| NTFS |  |
| 5 | Hiển thị đối tượng là cây thư mục con với trường hợp đối tượng là thư mục | FAT32 |  |
| NTFS |  |
|  | **Mức độ hoàn thành Project** | |  |

# Mô tả các bước thực hiện

## FAT32

### Tổng quan cấu trúc sourcecode

* Tên file: *FAT32.py*
* Các class chính:
  + ***Attribute****:* Chứa các thuộc tính của tập tin hoặc thư mục trong hệ thống tập tin FAT32. Định nghĩa một số flags như READ\_ONLY *(chỉ đọc)*, HIDDEN *(ẩn)*, SYSTEM *(hệ thống)*, VOLUME\_ID *(phân vùng)*, DIRECTORY *(thư mục),* ARCHIVE *(nén)* và LONG\_NAME *(tên dài)*

class Attribute(Flag):

    READ\_ONLY = 0x01

    HIDDEN = 0x02

    SYSTEM = 0x04

    VOLUME\_ID = 0x08

    DIRECTORY = 0x10

    ARCHIVE = 0x20

    LONG\_NAME = 0x0F

* + ***Status****:* Các trạng thái của tập tin hoặc thư mục trong hệ thống file tập tin FAT32. Định nghĩa các flags như DELETED *(đã xóa),* EMPTY *(trống),* NORMAL *(bình thường)*

class Status(Flag):

    DELETED = 0xE5

    EMPTY = 0x00

    NORMAL = 0xFF

* + ***BootSector****:* Phần boot sector của hệ thống tập tin FAT32. Phân tích và chứa các thông tin quan trọng chứa trong boot sector và cung cấp các hàm bổ sung như tính toán vị trí offset bắt đầu của bảng RDET và offset từ một cluster index cho trước

class BootSector:

    def \_\_init\_\_(self, data, name):

        self.oem\_name

        self.bytes\_per\_sector

        self.sectors\_per\_cluster

        self.reserved\_sectors

        self.fat\_count

        self.total\_sectors

        self.fat\_size

        self.root\_cluster

        self.volume\_label

        self.fat\_type

        self.boot\_code

        self.boot\_signature

        self.RDET\_start

    def offset\_from\_cluster(self, cluster\_idx)

    def \_\_str\_\_(self)

* + ***FAT****:* Đại diện cho Bảng Cấp phát Tệp *(File Allocation Table – FAT)* của hệ thống tệp FAT32. Phân tích và lưu trữ dữ liệu FAT, được sử dụng để theo dõi việc cấp phát và liên kết các cluster. Đồng thời cung cấp các phương thức để lấy chuỗi cluster *(cluster chain)* từ một cluster index cho trước

class FAT:

    def \_\_init\_\_(self, data):

        self.data

        self.size

        self.FAT

    def get\_cluster\_chain(self, start)

* + ***Entry****:* Đại diện cho mục tệp hoặc thư mục trong hệ thống tệp FAT32. Phân tích và lưu trữ các thuộc tính của một entry

class Entry:

    def \_\_init\_\_(self, data):

        self.name

        self.longFileName

        self.attr

        self.status

        self.reserved

        self.create\_time

        self.create\_date

        self.last\_access\_date

        self.last\_write\_time

        self.last\_write\_date

        self.starting\_cluster

        self.file\_size

        self.extension

    def \_\_str\_\_(self)

* + ***RDET****:* Đại diện cho Bảng Mục Thư mục Gốc *(Root Directory Entry Table – RDET)* của hệ thống tệp FAT32. Đọc và lưu trữ các entry trong RDET, chứa thông tin về các tệp và thư mục trong thư mục gốc. Cung cấp các hàm để tìm một mục cụ thể theo tên và lấy tổng số mục

class RDET:

    def \_\_init\_\_(self, pointer, offset, bytes\_per\_sector):

        self.entries

        self.size

        self.sector

self.read\_entries(pointer, bytes\_per\_sector)

    def read\_entries(self, pointer, bytes\_per\_sector=512)

    def find\_entry(self, name)

    def \_\_str\_\_(self)

def read\_chain(pointer, starting\_cluster, sectors\_per\_cluster, bytes\_per\_sector, fat, RDET\_start)

* + ***CDET****:* Đại diện cho Bảng Mục Thư mục Cụm *(Cluster Directory Entry Table – CDET)* của hệ thống tập tin FAT32. Đọc và lưu trữ các entry trong một CDET, chứa thông tin về các tập tin và thư mục trong một cluster cụ thể. Cung cấp các phương thức để tìm một entry cụ thể

class CDET:

    def \_\_init\_\_(self, data):

        self.entries

        self.read\_entries(data)

    def read\_entries(self, data)

    def find\_entry(self, name):

    def \_\_str\_\_(self)

* + ***Node****:* Đại diện cho một node trong cấu trúc cây thư mục của hệ thống tệp FAT32. Mỗi node tương ứng một tệp hoặc thư mục và chứa thông tin như node cha, node con, thông tin entry và đường dẫn thư mục. Cung cấp phương thức để đặt tên cho node

class Node:

    def \_\_init\_\_(self, dir = None, entry = None, isRoot = False):

        self.isRoot

        self.parent

        self.children

        self.info = entry

        self.dir = dir

    def setName(self, name)

* + ***FAT32****:* Đại diện cho toàn bộ hệ thống tệp FAT32. Khởi tạo các thành phần cần thiết như boot sector, RDET và node gốc. Cung cấp các phươngthức như duyệt cây thư mục, đọc nội dung của tệp .txt và hiển thị cấu trúc cây thư mục

class FAT32:

    def \_\_init\_\_(self, name):

        self.name

        self.ptr

        self.RDET

        self.root

    def offset\_from\_cluster(self, cluster\_index)

    def vis(self, start\_cluster, dir, parRoot = None)

    def get\_dir\_tree(self, start\_cluster, curRoot)

    def read\_txt\_file(self, txtNode)

    def draw\_dir\_tree(self, curNode, depth = 0)

### Đọc thông tin chi tiết của một phân vùng

* Khởi tạo: Khởi tạo đối tượng *BootSector* với các tham số *data* là dữ liệu 512 bytes đầu tiên và *name* là tên của phân vùng (*C:, D:,…*) được chọn, constructor của class là *\_\_init\_\_* sẽ được thực thi để phân tích 512 bytes *data* chứa thông tin của phân vùng *name:*

class BootSector:

    def \_\_init\_\_(self, data, name):

        self.oem\_name = data[3:11]

        self.bytes\_per\_sector = int.from\_bytes(data[11:13], byteorder='little')

        self.sectors\_per\_cluster = int.from\_bytes(data[13:14], byteorder='little')

        self.reserved\_sectors = int.from\_bytes(data[14:16], byteorder='little')

        self.fat\_count = int.from\_bytes(data[16:17], byteorder='little')

        self.total\_sectors = int.from\_bytes(data[32:36], byteorder='little')

        self.fat\_size = int.from\_bytes(data[36:40], byteorder='little')

        self.root\_cluster = int.from\_bytes(data[44:48], byteorder='little')

        self.volume\_label = name

        self.fat\_type = data[54:62]

        self.boot\_code = data[62:510]

        self.boot\_signature = data[510:512]

        self.RDET\_start = self.reserved\_sectors + self.fat\_count \* self.fat\_size

* Các giá trị, thông tin của phân vùng được đọc gồm có:

# Hình ảnh demo chương trình

# Nguồn tham khảo