**🙞🙞TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN🙜🙜**

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙣🕮🙡**

**BỘ MÔN HỆ ĐIỀU HÀNH**

ĐỒ ÁN 1: QUẢN LÍ HỆ THỐNG TẬP TIN TRÊN WINDOWS

* Giảng viên: Lê Viết Long
* Sinh viên thực hiện: Võ Hoàng Bảo Duy – 19127027

Trần Ngọc Lam – 19127040

Lê Minh Sĩ – 19127064

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Khoa Công nghệ Thông tin

Đại học Khoa học Tự nhiên TP HCM

Tháng 12/ 2020

Mục lục

[1. Thông tin chung 3](#_Toc75637589)

[a. Thông tin nhóm 3](#_Toc75637590)

[b. Bảng phân công công việc 3](#_Toc75637591)

[2. Thiết kế và xây dựng chương trình 4](#_Toc75637592)

[a. Phân tích các chức năng 4](#_Toc75637593)

[b. Xây dựng các chức năng 4](#_Toc75637594)

[3. Triển khai chương trình 8](#_Toc75637595)

[4. Tài liệu tham khảo 9](#_Toc75637596)

# Thông tin chung

## Thông tin nhóm

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ Tên** | **Email** |
| *19127027* | Võ Hoàng Bảo Duy | [19127027@student.hcmus.edu.vn](mailto:19127027@student.hcmus.edu.vn) |
| *19127040* | Trần Ngọc Lam | [19127040@student.hcmus.edu.vn](mailto:19127040@student.hcmus.edu.vn) |
| *19127064* | Lê Minh Sĩ | [19127064@student.hcmus.edu.vn](mailto:19127064@student.hcmus.edu.vn) |

## Bảng phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **Họ Tên** | **Phân công công việc** | **Mức độ hoàn thành** |
| 19127027 | Võ Hoàng Bảo Duy | * Đọc các thông tin được mô tả trong Boot Sector (FAT32). * Đọc và phân tích bảng RDET và bảng FAT. * Xây dựng cây thư mục cho hệ thống FAT32. | 100% |
| 19127040 | Trần Ngọc Lam | * Viết báo cáo. * Xây dựng giao diện hiển thị thông tin cho chương trình. | 100% |
| 19127064 | Lê Minh Sĩ | * Đọc các thông tin được mô tả trong Partition Boot Sector (NTFS). * Đọc và phân tích Master file table. * Xây dựng cây thư mục cho hệ thống NTFS. | 100% |
| Toàn bộ project | | | 100% |

# Thiết kế và xây dựng chương trình

## Phân tích các chức năng

* **FAT32:**
* Đọc Boot Sector FAT32 và lưu các thông số sau:
* Số byte trên mỗi sector
* Số sector trên mỗi cluster
* Sector bắt đầu bảng FAT, RDET, DATA.
* Lập bảng FAT:
* Đọc RDET
* Từ RDET gọi đệ quy đọc SDET
* Lập cây thư mục cho ổ đĩa từ RDET và SDET
* **NTFS:**
* Đọc Partition Boot Sector NTFS và lưu các thông số:
* OEM\_ID
* Thông số của BPB
* Extended BPB
* Sử dụng: Số byte / sector, Số sector / cluster và cluster bắt đầu của MFT để xác định offset của Master File Tabl
* Lập bảng Master File Table: đọc entry gồm:
* Entry Header
* Các attributes gồm: Attribute Header và Attribute
* Thông qua bảng Master File Table gọi đệ quy lập cây thư mục.

## Xây dựng các chức năng

* **FAT32:**
* Ta xây dựng class BootSectorFAT32 và PbrFat để đọc và lưu trữ các thông số của Boot Sector. Các thông số được được hiển thị của Boot Sector FAT32 bao gồm:

|  |  |
| --- | --- |
| **Tên** | **Giải thích** |
| BS\_jmpBoot | Mã lệnh để nhảy đến boot trap |
| BS\_OEM\_Name | Tên OEM dưới dạng kí tự |
| BPB\_BytesPerSec | Kích thước của Sector |
| BPB\_SecPerClus | Số sector trên mỗi cluster |
| BPB\_RsvdSecCnt | Số Sector chứa vùng dự trữ, trường này luôn hơn 0 |
| BPB\_NumFATs | Số lượng bản sao của FAT. Thông thường, giá trị của trường này là 2. |
| BPB\_RootEntCnt | Trong FAT12/16 thì trường này được sử dụng để chỉ số lượng Directory trong Root Directory. Trên FAT32 thì trường này không được sử dụng và nó luôn phải được set là 0. |
| BPB\_Media | Giá trị ngăn cách cố định bên của đĩa cứng, giá trị cố định chuẩn là 0xF8. |
| BPB\_SecPerTrack | Số Sector có trong 1 track |
| BPB\_NumHeads | Số lượng Head (đầu đọc) |
| BPB\_HiddSec | Số sector không thuộc ổ đĩa luân lý, tức là nằm phía trước địa chỉ bắt đầu cài đặt ổ đĩa luân lý |
| BPB\_TotalSec32 | Kích thước của phân vùng, được tính theo sectors. |
| BPB\_FATsz32 | Số Sector có trong 1 FAT luân lý |
| BPB\_ExtFlags | Bít 3～0: FAT active tính từ 0. Nếu bit số 7 là 1 thì trường này được sử dụng.  Bít 6～4: Dự trữ.Bit 7: bằng 0 thì toàn bộ FAT sẽ được mirroring. Nếu là 1 thì chỉ có FAT được chỉ định ở bit 3~0 là active thôi.  Bit 15～8: dự trữ. |
| BPB\_FSver | Version của FAT32. Byte cao là Major, Byte thấp là Minor. |
| BPB\_RootStartClus | Vị trí của Sector đầu tiên trong vùng Directory. |
| BPB\_FSInfo | Bên trong vùng dự trữ của FAT32 có vị trí Sector đặt cấu trúc FSINFO.　Thường là 1. Tức là Sector sau Boot Sector. |
| BPB\_BackupBootSec | Nếu khác giá trị 0, thì nó sẽ là giá của Sector chưa bản Backup của Boot Sector. |
| BPB\_Reserved | Trường dự phòng cho việc mở rộng trong tương lai. Tại thời điểm format ổ đĩa, chúng nên được gán là 0. |
| BS\_DriveNum | Số thứ tự ổ đĩa được sử dụng trong đĩa của BIOS IBM PC |
| BS\_Revervedl | Dự trữ (dùng cho WindowsNT), khi format thì cần set 0 cho giá trị này. |
| BS\_BootSig | Boot Signature mở rộng (0x29). Giá trị này sẽ quyết định sự tồn tại của 2\3 giá trị kế tiếp. |
| BS\_VolumeID | Số serial của ổ đĩa luân lý. |
| BS\_VolumLabel | Nhãn của ổ đĩa luân lý. |
| BS\_FileSysType | Loại FAT: FAT12, FAT16, FAT32 hoặc FAT. |

* Để thực hiện đọc bảng FAT, ta xây dựng class FatTable được định nghĩa trong boot\_sector\_fat32.py với các hàm chính: getRootDirectoryđể đọc RDET và getDirectory để đọc SDET.
* Hỗ trợ cho việc đọc bảng FAT và xây dựng cây thư mục sẽ có class Directory được định nghĩa trong directory\_fat32.py. Các hàm chính của đọc entry: getEntry, readAllEntry.
* Bên cạnh đó trong file\_entry\_fat32.py*,* hàm đọc entry chính readDirectoryShortEntrytrong class ShortFileNameEntry và hàm đọc entry phụ readDirectionLongEntry trong class LongFileNameEntry.
* Hỗ trợ cho việc xây dựng và hiển thị cây thư mục sẽ có class Node và class Root được định nghĩa trong directory\_fat32.py
* Các đặc tính của tập tin, thư mục được hiển thị bao gồm:
* Số lượng thư mục và tập tin (nếu là ổ đĩa)
* Đường dẫn tới file.
* Tên file.
* Thuộc tính của file: Read only, Hidden, System, Volume ID, Directory, Archive.
* Thời gian tạo file.
* Thời điểm điều chỉnh file.
* Lần truy cập file cuối cùng.
* Cluster đầu tiên
* Số lượng cluster
* Kích thước (bytes)
* Số lượng entry
* Sector bắt đầu và Sector kết thúc.
* **NTFS:**
* Ta xây dựng class BootSectorNTFS và Bpb (hàm \_init\_) để đọc và lưu trữ các thông số của Partition Boot Sector. Các thông số được được hiển thị của Partition Boot Sector NTFS bao gồm:

|  |  |
| --- | --- |
| **Tên** | **Giải thích** |
| oem\_id | Tên OEM dưới dạng kí tự |
| bytes\_per\_sector | Số byte trên mỗi sector |
| sectors\_per\_cluster | Số sector trên mỗi cluster |
| reserved\_sectors | Sector cho vùng dự trữ |
| media\_descriptor | Giá trị ngăn cách cố định bên của đĩa cứng |
| sectors\_per\_track | Số sector trên mỗi track |
| number\_of\_heads | Số lượng head (đầu đọc) |
| hidden\_sectors | Số sector không thuộc ổ đĩa luân lý, tức là nằm phía trước địa chỉ bắt đầu cài đặt ổ đĩa luân lý |
| total\_sectors | Kích thước của phân vùng, được tính theo sectors. |
| mft\_cluster | Số lượng cluster cho tệp $MFT |
| mft\_mirror\_cluster | Số lượng cluster cho tệp $MFTMirr |
| clusters\_per\_mft | Số lượng cluster trên mỗi phân đoạn của bản ghi |
| clusters\_per\_index | Số lượng cluster trên bộ đệm chỉ mục |
| volume\_serial | Số serial của ổ đĩa |
| checksum | Checksum |

* Để đọc entry, ta xây dựng class MFT với hàm get\_entry được định nghĩa trong partition\_boot\_sector\_ntfs.py.
* Cũng trong partition\_boot\_sector\_ntfs.py, ta có class MFTEntry (hàm load\_attributes và get\_attribute) để đọc Entry Header, Attribute Header và Attribute.
* Class MFTEntryHeader, hàm \_init\_ được dùng để đọc Entry Header.
* Class MRTAttrHeader, hàm \_init\_ được dùng để dọc Attribute Header.
* Class MFTAttr, hàm factory được dùng để đọc Attribute.
* Hỗ trợ cho việc xây dựng và hiển thị cây thư mục sẽ có class NodeNTFS và class RootNTFS được định nghĩa trong directory\_ntfs.py
* Các đặc tính của tập tin, tập mục được hiển thị bao gồm:
* Đường dẫn tới file.
* Tên file.
* Thuộc tính của file: Read only, Hidden, System, Volume ID, Directory, Archive.
* Thời gian tạo file.
* Thời điểm điều chỉnh file.
* Lần truy cập file cuối cùng.
* Sector bắt đầu và Sector kết thúc.

# Triển khai chương trình

1. **Giải thích các chức năng của chương trình**

* **Lưu ý:** Do chương trình xây dựng để đọc thư mục/tập tin trên USB, nên thuật toán xây dựng không tối ưu có thể dẫn đến dừng chương trình khi đọc các ổ đĩa có kích thước, số tập tin lớn
* Mục Thông tin chung:

Giới thiệu thông tin chung về đồ án, đề tài, sinh viên thực hiện đề tài và giáo viên hướng dẫn đề tài.

* Mục thông tin ổ đĩa:

Hiển thị danh sách các ổ cứng hiện thời, người sử dụng sẽ chọn ổ cứng cần xem và chọn “Boot Sector” để xem các thông tin về ổ cứng hoặc “Directory” để xem thông tin cây thư mục của ổ đĩa.

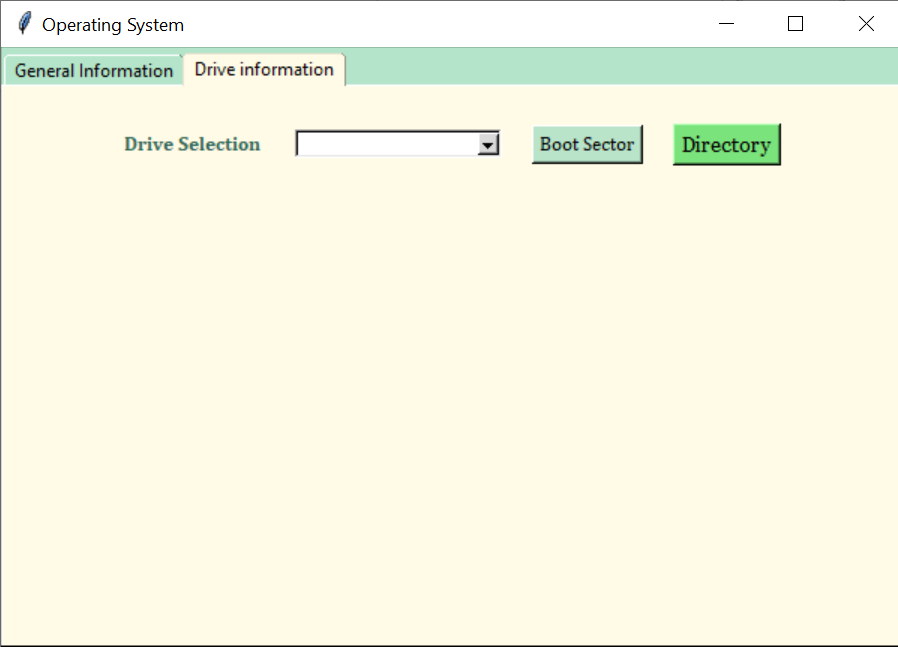
1. **Demo giao diện chương trình**

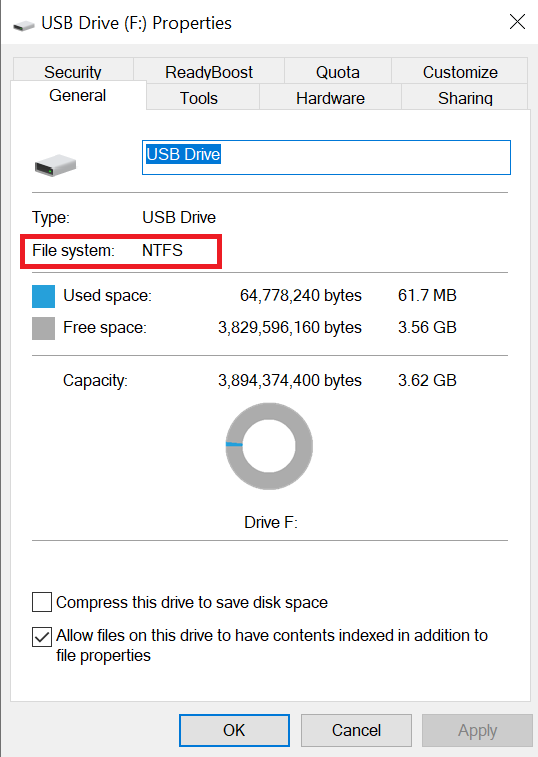
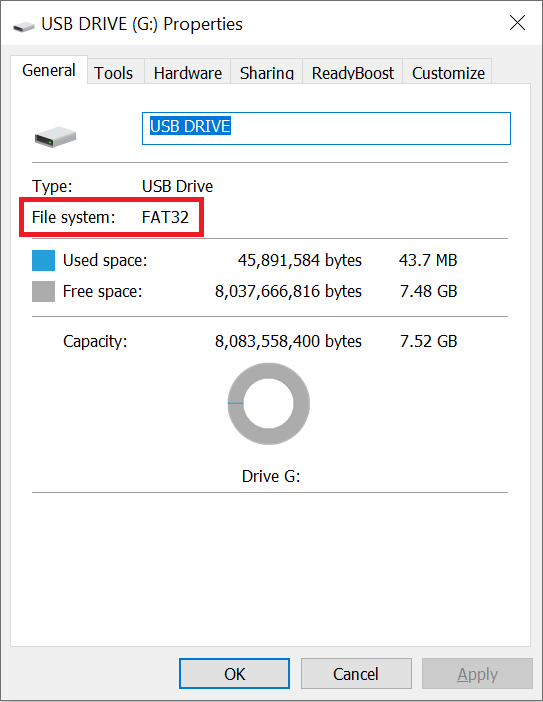
Chương trình được lập trình bằng ngôn ngữ Python.

* Mục Thông tin chung:
* Mục thông tin ổ đĩa:
* Mục thông tin ổ đĩa:

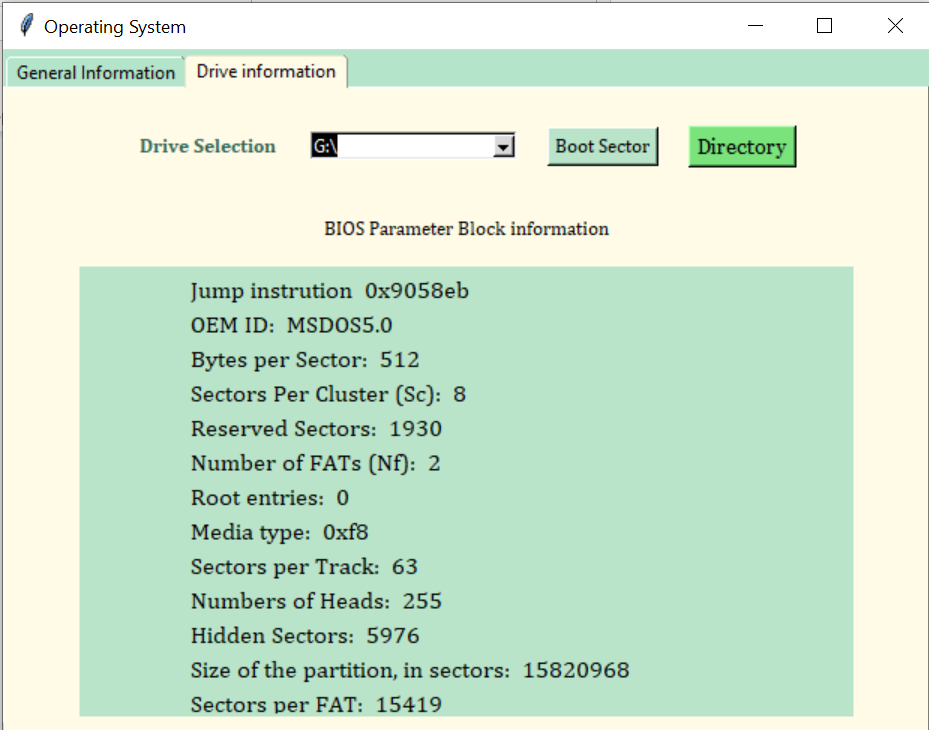
Ảnh có chứa văn bản

Mô tả được tạo tự động

* Mục thông tin ổ đĩa:

Ổ đĩa được sử dụng trong phần demo sẽ là ổ F (NTFS) có kích thước 4GB và ổ G (FAT32) có kích thước 16GB.

* **Xem Boot Sector:**

**FAT32:**

Ảnh có chứa văn bản

Mô tả được tạo tự động

**Ảnh có chứa văn bản

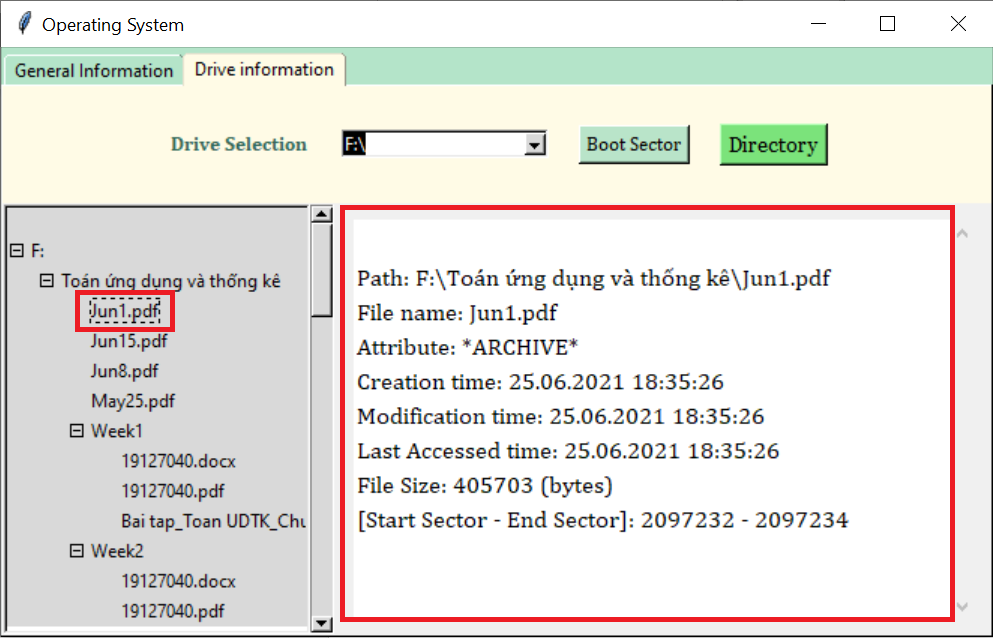
Mô tả được tạo tự độngNTFS:**

* **Xem cây thư mục:**
* Để xem đặc tính của tập tin/ thư mục: Nhấp chuột vào thư mục/ tập tin muốn xem, đặc tính của tập tin/ thư mục sẽ được hiển thị ở khung văn bản kế bên cây thư mục.
* Để mở tập tin trong cây thư mục: Nhấp đúp chuột vào tập tin muốn mở. Thao tác và quy tắc mở tập tin giống nhau ở cả hai hệ thống FAT32 và NTFS.
* Đối với tập tin có phần mở rộng là txt (text): chương trình sẽ tự mở một khung cửa sổ hiển thị nội dung tập tin.
* Đối với tập tin có phần mở rộng khác txt (text): chương trình sẽ hiện thông báo dùng phần mềm tương thích để đọc nội dung. Nếu người dùng chọn “Yes”, tập tin sẽ được mở bằng phần mềm ngoài tương thích hiện hành của máy tính. Nếu người dùng chọn “No”, tập tin sẽ không được mở.

**FAT32:** Xem cây thư mục của ổ đĩa G và đặc tính của tập tin “\_1\_intro.ppt”

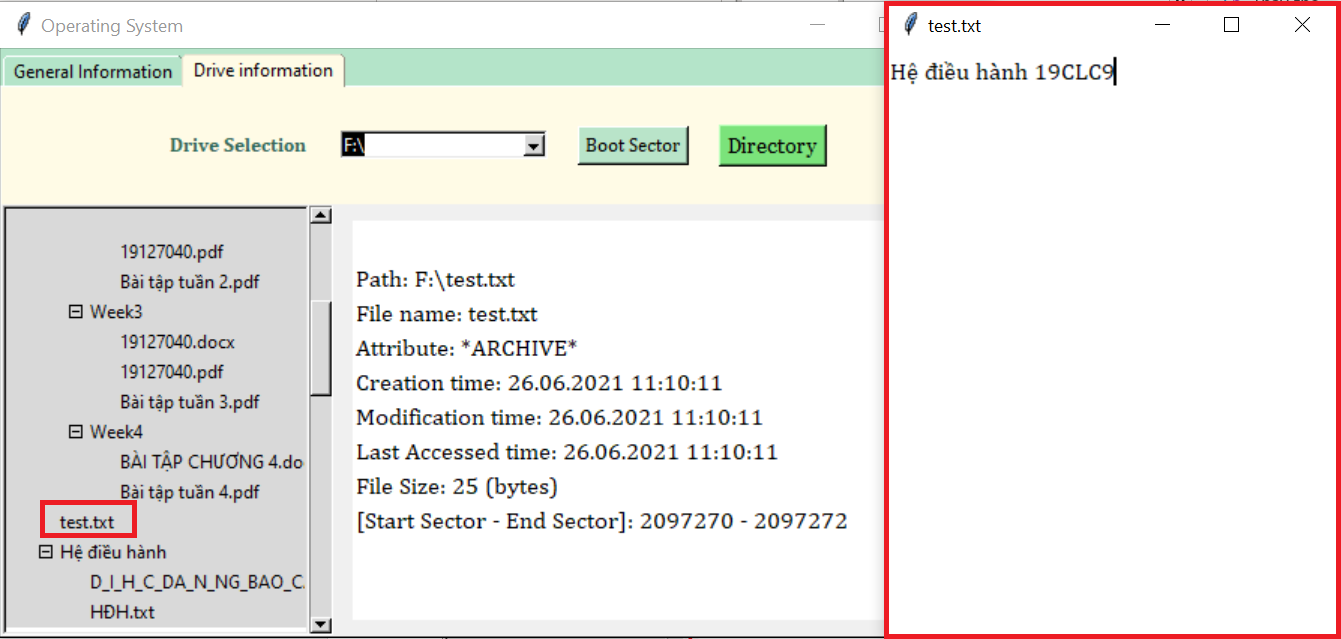
Ảnh có chứa văn bản

Mô tả được tạo tự động

**NTFS:** Xem cây thư mục của ổ đĩa F và đặc tính của tập tin “Jun1.pdf”

**Mở tập tin:**

*Thực hiện mở tập tin “test.txt” ở ổ đĩa F.*

**

*Ảnh có chứa văn bản

Mô tả được tạo tự độngThực hiện mở tập tin “HDH-07-Chap6.pdf” ở ổ đĩa F. Chọn “Yes” ở khung cửa sổ thông báo.*

Ảnh có chứa văn bản

Mô tả được tạo tự động*Tập tin được mở bằng Adobe Acrobat Reader DC*

Nhóm có hỗ trợ thêm video demo để có thể hiểu rõ hơn cách chạy của chương trình: [Hệ điều hành - Đồ án Quản lý hệ thống tập tin trên Windows - Demo - YouTube](https://www.youtube.com/watch?v=1J0Lw1sw0IE)

Đường dẫn tới Github của nhóm: [vhbaoduy/Project01\_OS (github.com)](https://github.com/vhbaoduy/Project01_OS?fbclid=IwAR0JPGyMo_8f_uimqLEGZOtXu5QtW0LknGYLO5tt0K9OVy9ayOO6pwy9ta0)

# Tài liệu tham khảo

<https://www.cse.scu.edu/~tschwarz/COEN252_09/Lectures/NTFS.html>

<http://ntfs.com/ntfs-partition-boot-sector.htm>

<https://pythonhosted.org/rawdisk/rawdisk.plugins.filesystems.ntfs.html>

<http://legiacong.blogspot.com/2014/04/he-thong-quan-ly-tap-tin-ntfs-6-mft-mft.html>

<https://legiacong.blogspot.com/2014/04/he-thong-quan-ly-tap-tin-ntfs-5-vbr-bpb.html?m=1>

Source tham khảo FAT32: <https://github.com/andrsv/Python-FAT-reader>

<https://lazytrick.wordpress.com/2015/12/27/khai-quat-ve-fat/>

<https://www.cse.scu.edu/~tschwarz/COEN252_09/Lectures/FAT.html>

UI:

[Python - GUI Programming (Tkinter) - Tutorialspoint](https://www.tutorialspoint.com/python/python_gui_programming.htm)

[Python Tkinter TreeView – How To Use - Python Guides](https://pythonguides.com/python-tkinter-treeview/)