

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN

HỆ ĐIỀU HÀNH – HỆ THỐNG QUẢN LÝ TẬP TIN

ĐỀ TÀI: FAT32 – NTFS

Giáo viên hướng dẫn:

ThS. Lê Viết Long

Thành phố Hồ Chí Minh, tháng 10 năm 2022

LỜI CẢM ƠN

Đầu tiên, nhóm em xin chân thành cảm ơn thầy ThS. Lê Viết Long, người đã hướng dẫn tận tình, giúp đỡ cho nhóm em hoàn thành đồ án này.

Bài tập đồ án lần này nhóm em đã cố gắng hết sức để hoàn thành nhưng cũng không tránh khỏi những sai sót. Vì thế em rất mong sự góp ý, chỉ bảo từ thầy cho em sau này có thể phát triển tốt hơn nữa.

Em rất mong nhận được sự quan tâm, giúp đỡ của thầy. Xin chân thành cảm ơn!

MỤC LỤC

1. Tổng quan	4
1.1. Thông tin thành viên trong nhóm	4
1.2. Bảng phân công công việc	4
1.3. Đánh giá đồ án	5
1.4. Thông tin cơ bản:	5
2. Mô tả các bước thực hiện	5
2.1. FAT32	5
2.1.1. Đọc thông tin Boot Sector	5
2.1.2. Đọc thông tin bảng FAT	7
2.1.3. Đọc cây thư mục gốc (RDET)	7
2.1.4. Đọc dữ liệu file	8
2.2. NTFS	8
2.2.1. Đọc thông tin Partition Boot Sector	8
2.2.2. Đọc MFT (Master File Table)	9
2.2.3. Đọc nội dung tập tin	10
3. Hướng dẫn chạy chương trình	10
3.1. Cài đặt	10
3.2. Hướng dẫn thực thi	10
4. Demo	13
4.1. FAT32:	13
4.2. NTFS:	18
Tài liệu tham khảo	25

1. Tổng quan

1.1. Thông tin thành viên trong nhóm

- Phan Phong Lưu – 20120326
- Phạm Bảo Huy – 20120298
- Ngô Võ Quang Huy – 20120295

1.2. Bảng phân công công việc

Thành viên Công việc	Ngô Võ Quang Huy 20120295	Phạm Bảo Huy 20120298	Phan Phong Lưu 20120326
Thiết kế cấu trúc chương trình			X
Đọc Boot Sector trong FAT32	X		
Đọc bảng FAT trong FAT32	X		
Đọc RDET trong FAT32			X
Đọc nội dung tập tin FAT32		X	
Đọc Partition Boot Sector trong NTFS	X		
Đọc Master File Table trong NTFS			X
Đọc nội dung tập tin NTFS		X	
Giao diện in ra màn hình		X	
Hoàn thiện			X

Kiểm thử		X	
Báo cáo	X		

1.3. Đánh giá đồ án

Yêu cầu	Mức độ hoàn thành (%)	Ghi chú
Đọc Boot Sector (phân vùng FAT32)	100	Hoàn thành
Đọc Partition Boot Sector (phân vùng NTFS)	100	Hoàn thành
Hiển thị cây thư mục của phân vùng FAT32	100	Hoàn thành
Hiển thị cây thư mục của phân vùng NTFS	100	Hoàn thành
Hiển thị thông tin nội dung tập tin txt và các loại khác mở bằng phần mềm tương thích.	100	Hoàn thành

- Đánh giá tổng thể mức độ hoàn thành: 100%.

1.4. Thông tin cơ bản:

- Ngôn ngữ lập trình: C++
- Code Editor: Visual Studio Code, version 1.72.2
- Môi trường lập trình: MingW-W64, version 12.1.0
- Các thư viện hỗ trợ:
 - Hỗ trợ giao diện: Windows.h
 - Xử lí file: fstream
 - Thuật toán: algorithm

2. Mô tả các bước thực hiện

2.1. FAT32

2.1.1. Đọc thông tin Boot Sector

- Boot sector là sector đầu tiên của phân vùng FAT32. Đọc các thông tin trong BPB (BIOS parameter block) và các thông tin khác.

Tên thuộc tính	Offset	Số Byte	Mô tả
BS_jmpBoot	0	3	Lệnh nhảy tới boot code
BS_OEMName	3	8	Tên OEM
BPB_BytsPerSec	11	2	Số byte trong 1 sector
BPB_SecPerClus	13	1	Số sector trong 1 cluster
BPB_RsvdSecCnt	14	2	Số sector dự trữ
BPB_NumFATs	16	1	Số lượng bảng FAT
BPB_RootEntCnt	17	2	Không dùng trong FAT32
BPB_TotSec16	19	2	Không dùng trong FAT32
BPB_Media	21	1	Lệnh điều khiển
BPB_FATSz16	22	2	Không dùng trong FAT32
BPB_SecPerTrk	24	2	Số sector trong 1 track
BPB_NumHeads	26	2	Số head (mặt đĩa)
BPB_HiddSec	28	4	Số sector ẩn
BPB_TotSec32	32	4	Tổng số sector
BPB_FATSz32	36	4	Kích thước bảng FAT
BPB_ExFlags	40	2	Cờ bổ sung
BPB_FS_Ver	42	2	Phiên bản FAT32
BPB_RootClus	44	4	Cluster đầu tiên của RDET
BPB_FSInfo	48	2	Số sector của FSINFO
BPB_BkBootSec	50	2	Chỉ dấu backup sector
BPB_Reserved	52	12	Dự trữ
BS_DrvNum	64	1	Số hiệu drive
BS_Reserved1	65	1	Dự trữ
BS_BootSig	66	1	Ký hiệu boot (0x29)
BS_VolID	67	4	Số seri ổ đĩa
BS_VolLab	71	11	Tên ổ đĩa
BS_FilSysType	82	8	Có giá trị “FAT32 ”
BootCode	90	420	
EndCode	510	2	Ký tự kết thúc 0x55AA

- Kiểm tra Boot sector vừa đọc có hợp lệ hay không: Kiểm tra BS_FilSysType là “FAT32 ” và hai byte cuối là 0x55AA.
- Các thông tin quan trọng

Thông tin	Giá trị	Mô tả
FirstFAT1Sector	BPB_RsvdSecC	Sector đầu tiên của bảng FAT thứ nhất
FirstFAT2Sector (nếu có)	BPB_RsvdSecCnt + BPB_FATSz32 (nếu BPB_NumFATs = 2) 0 (trường hợp còn lại)	Sector đầu tiên trong bảng FAT dự phòng nếu có

FirstDataSector	BPB_RsvdSecCnt+ (BPB_NumFATs * BPB_FATSz32)	Sector đầu tiên của cây thư mục gốc
-----------------	---	--

2.1.2. Đọc thông tin bảng FAT

- Vị trí sector đầu tiên của bảng FAT: FirstFAT1Sector.
- Cache thông tin bảng FAT: Vì khả năng lưu trữ của chương trình có giới hạn mà kích thước bảng FAT có thể rất lớn nên giải pháp của nhóm là cache một sector trong bảng FAT và lưu lại vị trí sector cache. Nếu thông tin cần đọc có trong cache thì tìm nội dung trong cache, nếu không thì chương trình sẽ load lại cache có chứa thông tin cần truy xuất.
- Xác định cluster kế tiếp của cluster hiện tại – nextClusterOf:
 - o Tìm sector chứa thông tin entry ứng với cluster:
 - $\text{sectorOffset} = \text{curCluster} / \text{EntryPerSector}$ ($\text{EntryPerSector} = \text{BytesPerSector} / 4$).
 - Load lại cache nếu sector chứa thông tin không nằm trong cache.
 - Tìm vị trí entry ứng với cluster trong sector.
 $\text{entryOffset} = (\text{curCluster} \% \text{EntryPerSector}) * 4$
 - Trả về cluster kế tiếp: 4 bytes từ entryOffset.

2.1.3. Đọc cây thư mục gốc (RDET)

- Cấu trúc dữ liệu Element lưu trữ thông tin file/folder đọc được

```
typedef struct Element {
    string name;
    string expandName;
    wstring longName;
    DWORD size;
    char type;
    WORD firstCluster;
} Element;
```

- Trong đó:

Kiểu dữ liệu	Tên thuộc tính	Mô tả
string	name	Tên chính
string	expandName	Tên mở rộng
wstring	longName	Tên dài (Tên hiển thị)
DWORD	size	Kích thước File
char	type	‘A’ nếu là File, ‘D’ nếu là folder
WORD	firstCluster	Cluster đầu tiên của dữ liệu

- Cây thư mục gốc sẽ bắt đầu từ FirstDataSector và chia thành các Entry. Trong FAT32, mỗi Entry có độ dài 32 bytes. Có hai loại Entry là Entry chính và Entry phụ.

Entry phụ bổ sung tên dài cho Entry chính, Entry chính đại diện cho một file/folder (Element). Cấu trúc của Entry chính:

Offset	Số byte	Thông tin
0	8	Tên chính của tập tin
8	3	Tên mở rộng
11	1	Thuộc tính
12	10	Không dùng
22	2	Giờ cập nhật tập tin
24	2	Ngày cập nhật tập tin
26	2	Cluster bắt đầu
28	4	Kích thước tập tin

- Khi đọc các entry chính kết hợp với entry phụ, ta sẽ được một tập tin nằm trong cây thư mục gốc. Nếu đó là folder, cluster bắt đầu của dữ liệu sẽ chứa các entry tương tự như cây thư mục gốc và các tập tin tại đây là con của folder này. Nếu đó là file, cluster bắt đầu sẽ chứa thông tin dữ liệu của file đó.
- Xác định điểm cluster kế tiếp: Vì dữ liệu của một tập tin có thể kéo dài hơn một cluster nên cần xác định cluster tiếp theo dựa trên nextClusterOf. Nếu tồn tại cluster kế tiếp thì nhảy tới cluster đó và tiếp tục đọc dữ liệu.

2.1.4. Đọc dữ liệu file

2.1.4.1. File txt

- Dựa trên thông tin firstCluster của Element, ta có thể tìm được cluster đầu tiên chứa dữ liệu của file txt, đọc và xuất ra màn hình. Nếu tồn tại cluster kế tiếp, đi tới cluster đó và lặp lại quá trình đọc tới khi không còn cluster kế tiếp nữa.

2.1.4.2. File khác

- Dựa trên đường dẫn đến file sẽ yêu cầu hệ thống mở file bằng ứng dụng mặc định.

2.2. NTFS

2.2.1. Đọc thông tin Partition Boot Sector

- Partition Boot Sector là sector đầu tiên của phân vùng NTFS, bao gồm các thông tin:

Tên thuộc tính	Offset	Số byte	Mô tả
Imp	0x00	3	Lệnh nhảy
Signature	0x03	8	Ký hiệu NTFS, có giá trị “NTFS ”
BytesPerSector	0x0B	2	Số byte trong 1 sector
SectorsPerCluster	0x0D	1	Số sector trong 1 cluster
ReservedSectors	0x0E	2	Dự trữ
Zeros1	0x10	3	Luôn bằng 0
NotUsed1	0x13	2	Không dùng
MediaDescriptor	0x15	1	Mô tả phương tiện

Zeros2	0x16	2	Luôn bằng 0
SectorsPerTrack	0x18	2	Số sector trong 1 track
NumberOfHeads	0x1A	2	Số head (mặt đĩa)
HiddenSectors	0x1C	4	Số sector ẩn
NotUsed2	0x20	4	Không dùng
NotUsed3	0x24	4	Không dùng
TotalSectors	0x28	8	Tổng số sector
LCN_MFT	0x30	8	Chỉ số cluster logic cho \$MFT đầu tiên
LCN_MFTMirr	0x38	8	Chỉ số cluster logic cho \$MFTMirr
ClustersPerIndexBlock	0x40	4	Số cluster cho 1 index block
VolumeSN	0x48	8	Số seri của phân vùng
Code	0x50	430	
EndCode	0x1FE	2	Ký hiệu kết thúc: 0x55AA

- Kiểm tra Partition Boot Sector có hợp lệ hay không: Signature = “NTFS ”
- Các thông tin quan trọng:

Thông tin	Giá trị	Mô tả
SectorSize	BytesPerSector	Số byte trong 1 sector
ClusterSize	SectorSize*BytesPerSector	Số byte trong 1 cluster
FileRecordSize	$2^{\lceil \log_2 \text{ClustersPerFileRecord} \rceil}$	Số byte của 1 \$MFT
MFTAddr	LCN_MFT * ClusterSize	Địa chỉ của \$MFT đầu tiên

2.2.2. Đọc MFT (Master File Table)

- MFT bắt đầu tại vị trí MFTAddr và được chia thành các \$MFT có kích thước FileRecordSize, thường là 1024 byte.
- Mỗi \$MFT sẽ chứa thông tin của một tập tin. Vùng MFT sẽ chứa toàn bộ tập tin, thư mục của toàn bộ phân vùng.
- Đọc thông tin của một \$MFT
- \$MFT bao gồm header của MFT và các Attribute
 - o Header: Bao gồm một số thông tin về \$MFT.
 - o Các Attribute: Một \$MFT bao gồm nhiều Attribute, mỗi attribute thể hiện một số thuộc tính của tập tin. Các Attribute được xếp nối tiếp nhau và tiếp sau MFT header.
- Đọc thông tin Header: 48 byte đầu tiên của \$MFT, trong đó có thông tin quan trọng là địa chỉ Attribute kế tiếp
- Đọc thông tin các Attribute: Đọc Attribute đầu tiên dựa trên địa chỉ tìm được ở header. Từ vị trí Attribute đầu tiên và kích thước của Attribute, có thể tìm được vị trí Attribute thứ hai và tiếp tục đọc hết các Attribute. Các thông tin có được thông qua đọc Attribute:
 - o Attribute \$STANDARD_INFORMATION: Loại tập tin và khả năng truy cập
 - o Attribute \$FILE_NAME: Tên tập tin và tham chiếu đến \$MFT cơ sở

- Attribute \$DATA: Kích thước phần nội dung của tập tin và vị trí chứa dữ liệu tập tin
- Cấu trúc dữ liệu lưu trữ tập tin:
- typedef struct tagElement {
 wstring name;
 ULONGLONG size;
 char type;
 DWORD ID;
 DWORD parentID;
 ULONGLONG dataOffset;
} Element;
- Trong đó:

Kiểu dữ liệu	Tên thuộc tính	Mô tả
wstring	name	Tên tập tin
ULONGLONG	size	Kích thước tập tin
char	type	'A' nếu là File, 'D' nếu là folder
DWORD	ID	Số thứ tự của \$MFT
DWORD	parentID	Số thứ tự của \$MFT cha
ULONGLONG	dataOffset	Vị trí chứa dữ liệu

- Mỗi tập tin có một ID duy nhất là số thứ tự của \$MFT. Các tập tin trên thư mục gốc sẽ có cùng parentID và được đưa về 0. Các tập tin khác sẽ có parentID là ID của thư mục cha của tập tin đó.

2.2.3. Đọc nội dung tập tin

2.2.3.1. Tập tin txt

- Dựa trên dataOffset, ta tìm được vị trí bắt đầu dữ liệu của tập tin. Đối với Attribute \$DATA_RESIDENT, dữ liệu nằm ngay trên \$MFT và dataOffset là vị trí dữ liệu của Attribute \$DATA_RESIDENT. Đối với Attribute \$DATA_NONRESIDENT cần đọc nội dung của Attribute này và decode ra vị trí bắt đầu của dữ liệu

2.2.3.2. Tập tin khác

- Dựa trên đường dẫn đến file sẽ yêu cầu hệ thống mở file bằng ứng dụng mặc định.

3. Hướng dẫn chạy chương trình

3.1. Cài đặt

- Các loại phần mềm cần cài đặt để chạy chương trình:
 - Code Editor: Visual Studio Code, Sublime Text,...
 - C++ compiler (g++)
 - Các thư viện được dùng trong source code.
- Tải về các file source code để chạy chương trình.

3.2. Hướng dẫn thực thi

- Đầu tiên, mở file main.cpp trong ứng dụng Code Editor đã cài đặt.



```
140 int main(int argc, char*argv[]) {
141     _setmode(_fileno(stdout), _O_WTEXT);
142     _setmode(_fileno(stdin), _O_WTEXT);
143
144     wstring RootDir = L"";
145     string disk = readInput(RootDir);
146     int BS_flag = BS.load(disk);
147     if (BS_flag = 1 || BS_flag = 3) {
148         wcout << L"This disk is unreadable or not exist.\n";
149         system("pause");
150         return 1;
151     }
152     else if (BS_flag = 2) {
153         wcout << L"This disk is not formatted as FAT32.\n";
154         system("pause");
155         return 1;
156     }
157     else if (BS_flag != 0) {
158         wcout << L"Something went wrong while reading this disk.\n";
159         system("pause");
160         return 1;
161     }
162     BS.print();
163     FAT.load(&BS);
164     DT.init(&FAT);
165
166     dir.push_back(RootDir);
167     DT.readDirectoryTree(2);
168     DT.printList();
169
170     while (readCommand());
171     return 0;
172 }
173
174
```

- Tiếp theo, ta sẽ build file main.cpp.

Hệ điều hành – Hệ thống quản lý tập tin

```
FAT_NTFS > C++ main.cpp > readCommand()
70
71     do {
72         getline(wcin, input, L'\n');
73     } while (input.length() <= 2);
74     wcin.ignore(1000, '\n');
75
76     wstringstream wiss(input);
77     vector<wstring> commands;
78     wstring command = L"";
79     bool isContinous = false;
80     for (auto c : input)
81     {
82         if (c == L' ') {
83             isContinous = !isContinous;
84             continue;
85         }
86         else if (c == L' ' && !isContinous)
87         {
88             commands.push_back(command);
89             command = L"";
90         }
91         else {
92             command = command + c;
93         }
94     }
95     commands.push_back(command);
96 }
```







PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

* Executing task: C/C++: g++.exe build active file

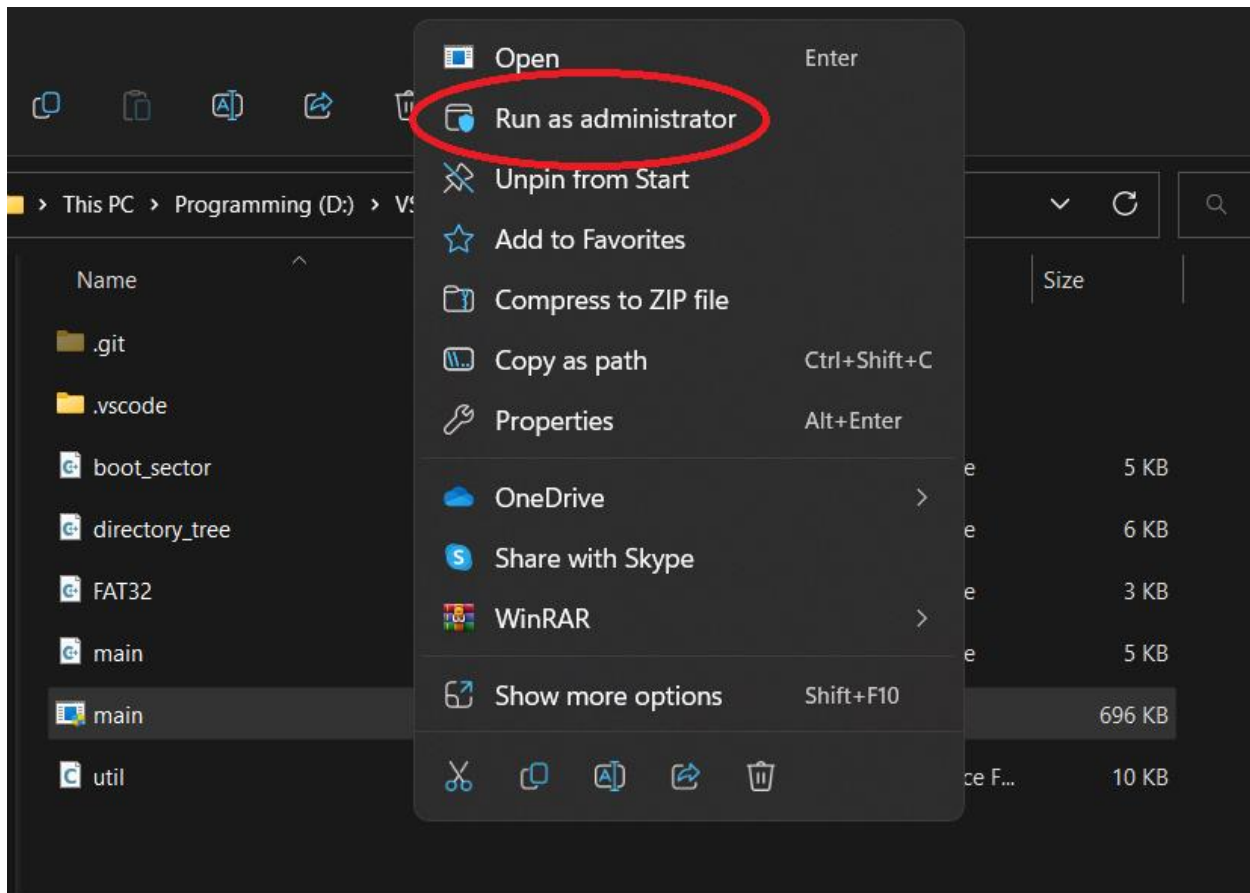
Starting build...
C:\msys64\mingw64\bin\g++.exe -fdiagnostics-color=always -g "D:\VS Code\FAT\TEST\WorkWithFAT\FAT_NTFS\main.cpp" -o "D:\VS Code\FAT\TEST\WorkWithFAT\FAT_NTFS\main.exe"

Build finished successfully.
* Terminal will be reused by tasks, press any key to close it.

- Sau khi build thành công, folder chứa file main.cpp sẽ xuất hiện file .exe.

	boot_sector	10/20/2022 12:12 PM	C++ Source File	5 KB
	directory_tree	10/20/2022 12:12 PM	C++ Source File	6 KB
	FAT32	10/20/2022 12:12 PM	C++ Source File	3 KB
	main	10/20/2022 12:12 PM	C++ Source File	5 KB
	main	10/20/2022 12:12 PM	Application	696 KB
	util	10/20/2022 12:12 PM	C Header Source F...	10 KB

- Để chạy chương trình, ta sẽ cho file .exe chạy với quyền admin, bằng cách nhấp chuột phải vào file .exe, rồi chọn *Run as administrator* để chạy chương trình.

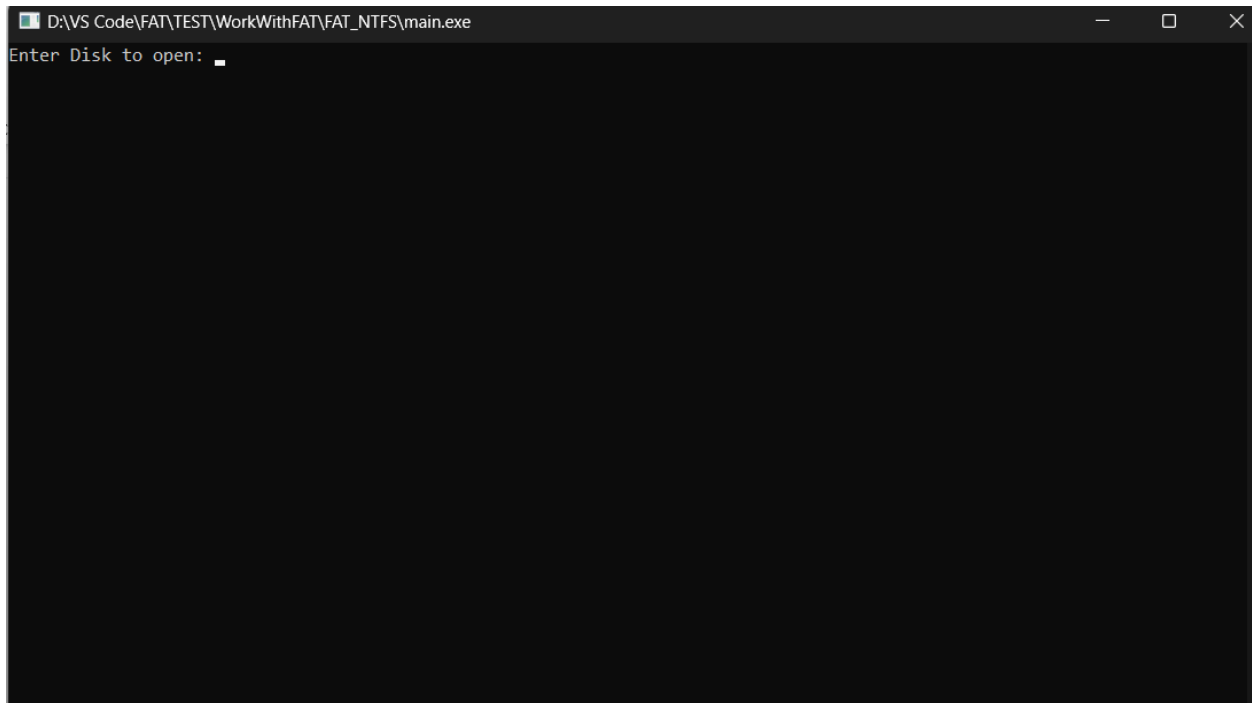


- Thế là chúng ta đã có thể vào chương trình.

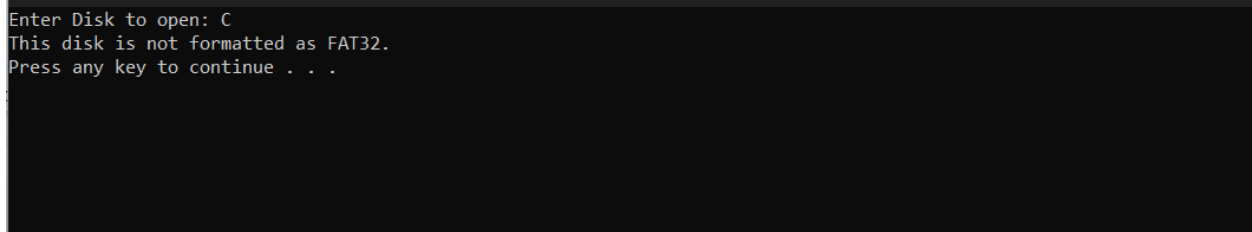
4. Demo

4.1. FAT32:

- Giao diện ban đầu khi bắt đầu chương trình.



- Yêu cầu nhập ổ đĩa muốn mở (ổ đĩa nhập vào bắt buộc là phân vùng FAT32).
 - Nếu ổ đĩa không thoả, xuất thông báo và thoát chương trình.



- Nếu thoả, chương trình xuất thông số Boot Sector và truy cập vào ổ đĩa:

```
Enter Disk to open: F
Jump Boot Code:      EB 58 90
Version/OS:          MSDOS5.0
Bytes per Sector:    512
Sectors per Cluster: 32
Reserved Sector Count: 314
Number FATs:         2
Root Entry Count:    0
Total Sector 16:     0
Media:              248
FAT16 size:         0
Sectors per Track:   63
Number Heads:       255
Hidden Sector:      834138112
Total Sector 32:    66494464
FAT Size 32:        16227
Extra Flags 32:     0
FAT32 Version:      0
Root Cluster:       2
FS Info:            1
BK Boot Sector:     6
Reserved:           00 00 00 00 00 00 00 00 00 00 00 00
Driver Number:      128
Reserved 1:         0
Boot Signature:     41
Volume ID:          -1297428608

Name                               Size                Type    firstCluster
-----
Kiểm tra                          0                  D        11
Test01                            0                  D        10
F:\>
```

- Mỗi folder hoặc file sẽ xuất hiện ba thuộc tính: Size (mặc định là 0 đối với folder), Type – loại D (Directory), loại A (Archive), firstCluster – vị trí Cluster đầu tiên của file hoặc folder.
- Gõ *cmdlst* để biết thêm về danh sách các lệnh để thao tác trong chương trình:

```
F:\> cmdlst
*****
cd <dir>    : go to <dir>
cd ..      : go to parent directory
open <file>: catch .txt file or open other file
quit       : quit the program
*****
F:\> █
```

- Truy cập vào folder *Test01*, nhập lệnh: *cd Test01*

```
Enter Disk to open: F
```

Name	Size	Type	firstCluster
Kiểm tra	0	D	11
Test01	0	D	10

```
F:\>
```

Name	Size	Type	firstCluster
Content	0	D	14
dddd.png	627061	A	22
image01.jpg	21977	A	12

```
F:\Test01\>
```

- Chúng ta có thể phân biệt được file với folder thông qua Type hoặc màu sắc. Nếu là folder thì tên sẽ có màu vàng và Type là D. Còn nếu là file thì tên sẽ có màu xanh và Type là A.
- Tiếp theo, truy cập vào Content. Gõ lệnh: *cd Content*

Name	Size	Type	firstCluster
Content	0	D	14
dddd.png	627061	A	22
image01.jpg	21977	A	12

```
F:\Test01\> cd Content
```

Name	Size	Type	firstCluster
image02.jpg	94193	A	16
number.xlsx	8460	A	63

```
F:\Test01\Content\>
```

- Vào được folder *Content*, ta sẽ mở thử lần lượt 2 file *image02.jpg* và *number.xlsx*.
Cú pháp: *open + filename*.

Name	Size	Type	firstCluster
image02.jpg	94193	A	16
number.xlsx	8460	A	63

```
F:\Test01\Content\> open image02.jpg  
F:\Test01\Content\> open number.xlsx
```

- Sau khi gõ từng dòng lệnh *open* thì từng file sẽ mở ra.

- Quay lại thư mục cha bằng câu lệnh: `cd ..`

Name	Size	Type	firstCluster
image02.jpg	94193	A	16
number.xlsx	8460	A	63

```
F:\Test01\Content\> cd ..
```

Name	Size	Type	firstCluster
Content	0	D	14
dddd.png	627061	A	22
image01.jpg	21977	A	12

```
F:\Test01\>
```

- Tiếp theo, ta quay lại thư mục gốc, sau đó truy cập vào folder *Kiểm tra*.

```
Enter Disk to open: F
```

Name	Size	Type	firstCluster
Kiểm tra	0	D	11
Test01	0	D	10

```
F:\> cd "Kiểm tra"
```

Name	Size	Type	firstCluster
That's Hilarious.txt	9788	A	15

```
F:\Kiểm tra\>
```

- Tiếp theo, ta sẽ thử mở file *That's Hilarious.txt* trong folder *Kiểm tra*. Gõ cú pháp: `open + filename`.

Name	Size	Type	firstCluster
That's Hilarious.txt	9788	A	15

```
F:\Kiểm tra\> open "That's Hilarious.txt"
```

- Nội dung của file txt sẽ không cần dùng ứng dụng thứ ba để xem mà nội dung được hiển thị trực tiếp trên giao diện chương trình.

Name	Size	Type	firstCluster
-----	-----	-----	-----
That's Hilarious.txt	9788	A	15

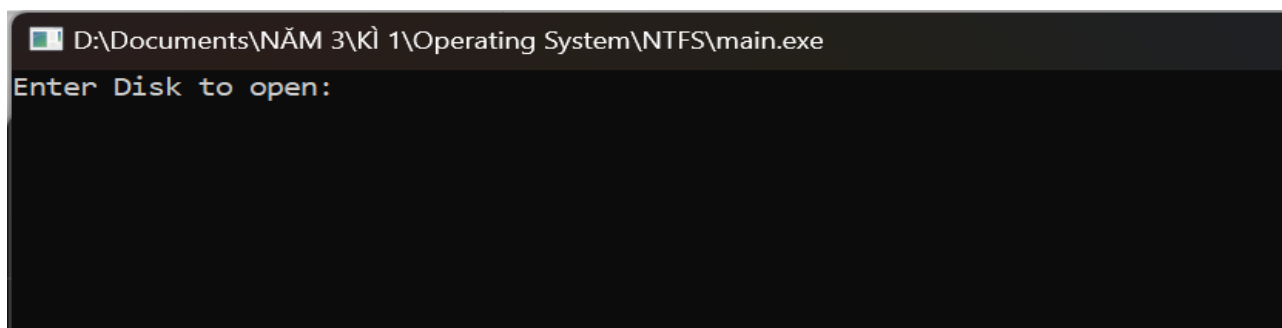
```
F:\Kiểm tra\> open "That's Hilarious.txt"
Look how all the tables
Look how all the tables have turned
Guess you're finally realizin' how bad you messed it up
Girl, you're only makin'
Girl, you're only makin' it worse
When you call like you always do when you want someone
You took away a year
Of my fuckin' life and I can't get it back no more
So when I see those tears
Comin' out your eyes, I hope it's me they're for
You didn't love when you had me
But now you need me so badly
You can't be serious
That's hilarious
Thinkin' I would still want you
After the things you put me through
Yeah, you're delirious
That's hilarious
Now you put the blame in (now you put the blame in)
Now you put the blame in reverse
Tryna make me feel guilty for everything you've done
You're another lesson (you're another lesson)
You're just another lesson I learned
Don't give your heart to a girl who's still got a broken one
```

- Cuối cùng, câu lệnh *quit* sẽ giúp ta thoát chương trình.

```
That's Hilarious (That's Hilarious)
F:\Kiểm tra\> quit_
```

4.2. NTFS:

- Giao diện ban đầu khi bắt đầu chương trình.



- Yêu cầu nhập ổ đĩa muốn mở (ổ đĩa nhập vào bắt buộc là phân vùng NTFS).
 - o Nếu ổ đĩa không thoả, xuất thông báo và thoát chương trình.

```
D:\Documents\NĂM 3\Kì 1\Operating System\NTFS\main.exe
Enter Disk to open: E
Not NTFS
Press any key to continue . . .
```

- Nếu thoả, sẽ truy cập vào ổ đĩa:

```
D:\Documents\NĂM 3\Kì 1\Operating System\NTFS\main.exe
Enter Disk to open: F
Sector size:          512 bytes
Cluster size:        4096 bytes
File Record size:    1024 bytes
Index Block size:    4096 bytes
MFT Address:         0x00000000C0000000
.
Name                  Size          Type          MFT Entry
-----
Files                  0            D             50
Presentations          0            D             43
mã nguồn              0            D             51
bg.jpg                6723941     A             742
demo.txt              8304699     A             42
F:\>
```

- Mỗi folder hoặc file sẽ xuất hiện ba thuộc tính: Size (mặc định là 0 đối với folder), Type – loại D (Directory), loại A (Archive), MFT Entry – số thứ tự của MFT Entry trong MFT (Master File Table).
- Gõ *cmdlst* để biết thêm về danh sách các lệnh để thao tác trong chương trình:

```
F:\> cmdlst
*****
cd <dir>      : go to <dir>
cd ..        : go to parent directory
open <file>:  catch .txt file or open other file
quit         : quit the program
*****
F:\> █
```

- Thử mở file *bg.jpg* trong ổ đĩa, nhập lệnh: *open bg.jpg*

Hệ điều hành – Hệ thống quản lí tập tin

Name	Size	Type	MFT Entry
Files	0	D	50
Presentations	0	D	43
mã nguồn	0	D	51
bg.jpg	6723941	A	742
demo.txt	8304699	A	42

```
F:\> open bg.jpg
```

- Tập tin sẽ được hiển thị trên màn hình như hình bên dưới



- Truy cập vào folder *Files*, nhập lệnh: `cd Files`

```
D:\Documents\NĂM 3\Kì 1\Operating System\NTFS\main.exe
```

```
Enter Disk to open: F
```

Sector size: 512 bytes
Cluster size: 4096 bytes
File Record size: 1024 bytes
Index Block size: 4096 bytes
MFT Address: 0x00000000C0000000

Name	Size	Type	MFT Entry
Files	0	D	50
Presentations	0	D	43
mã nguồn	0	D	51
bg.jpg	6723941	A	742
demo.txt	8304699	A	42

```
F:\> cd Files
```

Name	Size	Type	MFT Entry
books	0	D	39
Demo.csv	2	A	743
rarreg.key	476	A	52

F:\Files\>

- Chúng ta có thể phân biệt được file với folder thông qua Type hoặc màu sắc. Nếu là folder thì tên sẽ có màu vàng và Type là D. Còn nếu là file thì tên sẽ có màu xanh và Type là A.
- Tiếp theo, truy cập vào folder books. Gõ lệnh: *cd books*

Name	Size	Type	MFT Entry
books	0	D	39
Demo.csv	2	A	743
rarreg.key	476	A	52

F:\Files\> cd books

Name	Size	Type	MFT Entry
test1.pdf	22966711	A	41
test2.pdf	23842949	A	40

F:\Files\books\> █

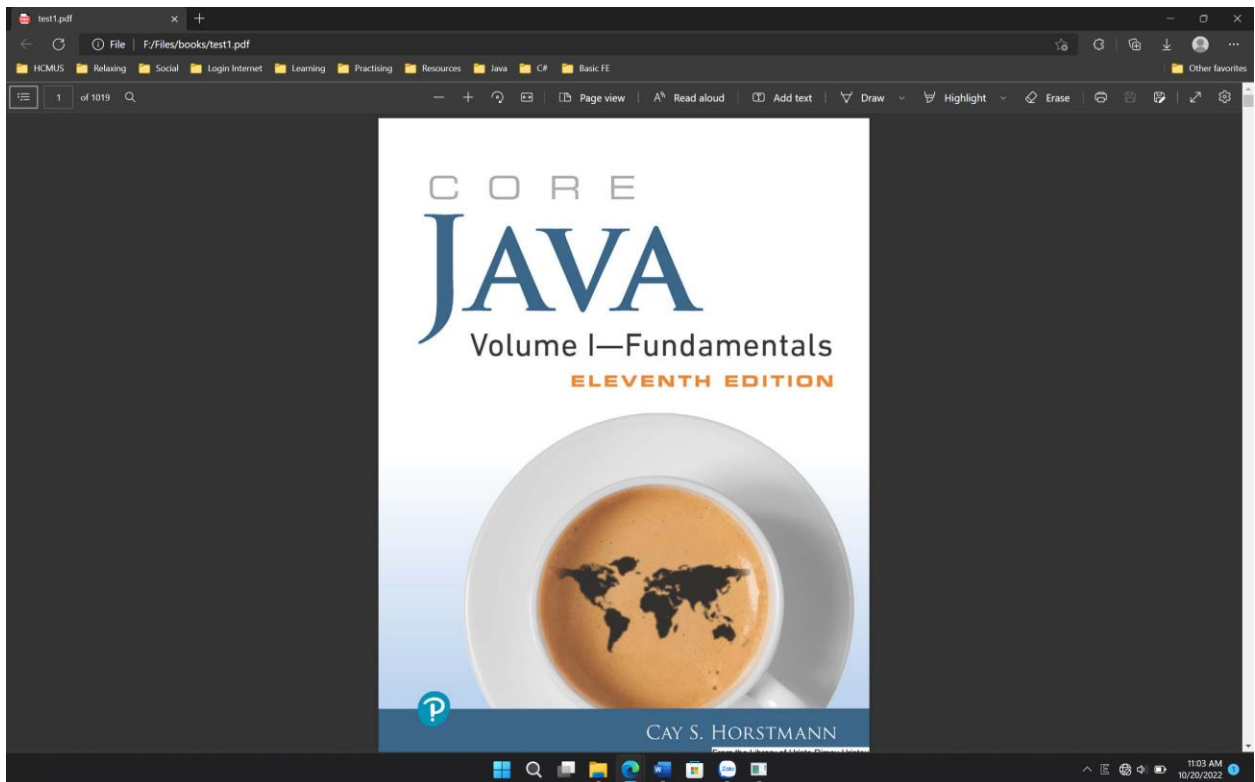
- Vào được folder books, ta sẽ mở thử file *test1.pdf*.

Name	Size	Type	MFT Entry
test1.pdf	22966711	A	41
test2.pdf	23842949	A	40

F:\Files\books\> open test1.pdf

- Tập tin sẽ được hiển thị trên màn hình như hình bên dưới

Hệ điều hành – Hệ thống quản lý tập tin



- Quay lại thư mục cha bằng câu lệnh: `cd ..`

```
F:\Files\books\> cd ..
Name                                     Size      Type      MFT Entry
-----
books                                   0          D         39
Demo.csv                               2          A        743
rarreg.key                             476        A         52
F:\Files\>
```

- Tiếp tục quay lại thư mục gốc bằng câu lệnh: `cd ..`

```
F:\Files\> cd ..
Name                                     Size      Type      MFT Entry
-----
Files                                   0          D         50
Presentations                           0          D         43
mã nguồn                               0          D         51
bg.jpg                                 6723941    A        742
demo.txt                               8304699    A         42
F:\>
```

- Tiếp theo, ta quay lại thư mục gốc, sau đó truy cập vào folder *mã nguồn*.

Name	Size	Type	MFT Entry
Files	0	D	50
Presentations	0	D	43
mã nguồn	0	D	51
bg.jpg	6723941	A	742
demo.txt	8304699	A	42

```
F:\> cd "mã nguồn"
```

Name	Size	Type	MFT Entry
gitlog.txt	85132	A	53

```
F:\mã nguồn\> _
```

- Tiếp theo, ta sẽ thử mở file *gitlog.txt* trong folder mã nguồn.

Name	Size	Type	MFT Entry
gitlog.txt	85132	A	53

```
F:\mã nguồn\> open gitlog.txt
```

- Nội dung của file txt sẽ không cần dùng ứng dụng thứ ba để xem mà nội dung được hiển thị trực tiếp trên giao diện chương trình.

Name	Size	Type	MFT Entry
gitlog.txt	85132	A	53

```
F:\mã nguồn\> open gitlog.txt
* commit c4c7e42c4477da8cefbe227906127fab9d43c3c0
  Author: 20120298 <pbhuy0909@gmail.com>
  Date:   Sun Oct 2 21:05:52 2022 +0700

    save file and submit

diff --git a/index.html b/index.html
index 0ff830e..f9b4a74 100644
--- a/index.html
+++ b/index.html
@@ -42,21 +42,29 @@
     </div>
   </li>
   <li class="side-box">
     <div class="tittle-sidebar"><span>NEWS 02</span></div>
+    <div class="tittle-sidebar">
+      <span class="ui-icon ui-icon-arrowthick-2-n-s"></span>
+      <span>NEWS 02</span></div>
     <div class="content">
       <p>content content content content content <br>
         content content content content content</p>
     </div>
   </li>
   <li class="side-box">
     <div class="tittle-sidebar"><span>NEWS 03</span></div> [.. +84108 characters]
```

- Cuối cùng, câu lệnh *quit* sẽ giúp ta thoát chương trình.

Name	Size	Type	MFT	Entry

gitlog.txt	85132	A		53
F:\mã nguồn\> quit				

Tài liệu tham khảo

- (1) Lê Gia Công, blogspot, từ <http://legiacong.blogspot.com/>.
- (2) Microsoft FAT Specification Document, Massachusetts Institute of Technology.
- (3) FAT32 Boot Sector, Locating Files and Dirs, Florida State University, từ https://www.cs.fsu.edu/~cop4610t/lectures/project3/Week11/Slides_week11.pdf
- (4) Review NTFS Basics, Australian Journal of Basic and Applied Sciences
- (5) NTFS, từ <https://www.ntfs.com/index.html>
- (6) Pontifical Catholic University of Peru, NTFS, từ http://inform.pucp.edu.pe/~inf232/Ntfs/ntfs_doc_v0.5/index.html