Programmer's Manual: Module 6

Oreo Team

Class List:

Here are the classes, structs, unions, and interfaces:

- directoryEntry
- bootSector

directoryEntry:

#include <stdio.h>

Public Attributes:

char fileName

char ext

int attribute

int fileSize

int firstCluster

uint16_t reserved

uint16_t creationTime

uint16_t creationDate

uint16_t lastAccessDate

uint16_t ignore

uint16_t lastWriteTime

uint16_t lastWriteDate

bootSector:

#include <stdio.h>

Public Attributes:

uint8_t ignore

uint16_t bytesPerSector

uint8_t sectorsPerCluster

uint16_t numOfReservedSectors
unint8_t numOfFatCopies
uint16_t maxNumberOfRootDirect
uint16_t totalNumOfSectors
uint8_t ignore2
uint16_t numOfSectorsPerFat
uint16_t sectorsPerTrack
uint16_t numOfHeads
uint32_t numOfHiddenSectors
uint32_t totalSectorCount
uint8_t ignore3
uint32_t totalSCforF32
uint8_t ignore4
uint8_t bootSignature
uint32_t volumeID
unsigned char volumeLabel
unsigned char fileSysType
uint8_t ignore5

File List:

- r6.c
- r6.h

r6.c File Reference:

```
#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
#include “r6.h”
#include <string.h>
int FAT[4086]
int directory
```

char current_directory[]

int quit

char current_path

Functions:

int getInt(int sizeOfbytes);

param: sizeOfByte – bytes you want to convert

return: int – the converted byte to int type

This function converts the bytes to int type.

void moveToSector(int sector);

param: sector – sector you want to move to

return: void

This function moves to the sector given.

void printRoot(void);

param: void

return: void

This function calls the print root directory function.

void printDir(directoryEntry* p, int num);

param: p – pointer to entry

param: num – number of entries in directory

return: void

This function prints the files and subdirectory in current directory.

void printEntry(directoryEntry entry);

param: entry – the entry to the directory

return: void

This function prints an entry's attributes.

void Type(char* name, char* ext);

param: name – name of file

param: ext – extention of file

return: void

This function prints a file when given the name and extention.

void init();

param: void

return: void

This function initializes the boot and root directory structures.

int numberOfSectors(int startingSector);

param: startingSector – the beginning sector int

return: int – number of sectors

This function gets the number of sectors from the starting sector.

void locateDirectory(directoryEntry* directory,int numberOfentries,int firstSector);

param: directory – current directory pointer

param: numberOfEntries – number of entries to cycle through

param: firstSector – the first sector int

return: void

This function searches for a directory by moving through sectors.

void interface(FILE*);

param: file – the file pointer to the disk image

return: void

This function is the main interface, calls the command line for user input.

void command_line(FILE*);

param: file – the file pointer to the disk image

return: void

This function is the command line for the user input.

void help();

param: void

return: void

This function presents a list of commands and their use to the user.

void printBootSector();

param: void

return: void

This function prints the boot sector.

int renameFile(File *, const char *old , const char * new);

param: file – the file pointer to the disk image

param: old – the original name of the file

param: new – the new name of the file

This function renames a file.

int Equals(const char *str1, const char * str2);

param: str1 – filename string

param: str2 – filename string

return: int – if matches or not

This function compares two filenames.

void change_directory(char* name);

param: name – directory name

return: void

This function changes the current directory.

void list_directory();

param: void

return: void

This function lists everything in the current directory.

r6.h File Reference:

```
FILE* fpointer;
int fatTale;
directoryEntry root[224];
directoryEntry* curr;
int sizeOfCurr;
int startOfCurr;
#define color_red "\x1b[31m"
#define color_green "\x1b[32m"
#define color_blue "\x1b[34m"
#define color_yellow "\x1b[33m"
#define color_clear "\x1b[0m"
```

Structs & Typedefs:

bootSector

directoryEntry