Junxiang Wen

Edit by 11/13/2019

3207

Project 4A: Pseudocode for [Simple File System (Project 4)](https://templeu.instructure.com/courses/63780/assignments/573337)

1. Partition of the whole virtual disk file
2. A graph to show the partition

b. Size of the virtual disk file

For whole disk: 65536 Kb

For data file: 32768 Kb

c. Total number of blocks

16384

d. Structs to illustrate the detailed structure

the virtual disk has 16,384 blocks total and each block holds 4 Kb

Maximum file number: 256

Total of file fize: 32768Kb

Block for super block: 1

Block for data file: 8192

Block for iNode: 8191

1. General design illustration
2. Use inode or FAT

inode

1. How are blocks allocated to files

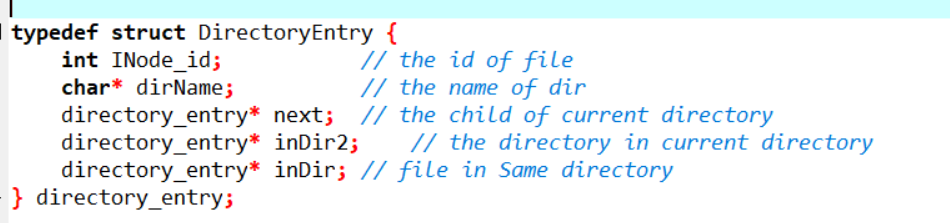
Each file have a inode, we can block array in inode struct. The first block will be in block[0]. And store the data in block into disk immediately

1. How to assemble files which are larger than one block

If the data is larger than one block, we can disrupte the data into serval sections. For example, we have a data is 3000 byte. It can disrupte into 1024+1024+952

1. How to distinguish normal files from directory files

The root should be an directory files. We can make a struct such like below



1. How to keep track of free spaces

Keep tracking the next free spaces, whenever we use or unused the free space

f. Where to accommodate the root directory

first block is superblock and it can store the root directory.

1. Data structures

In file.h

4 Pseudocode

In file.c

5. Testing

a. Estimation of possible bugs

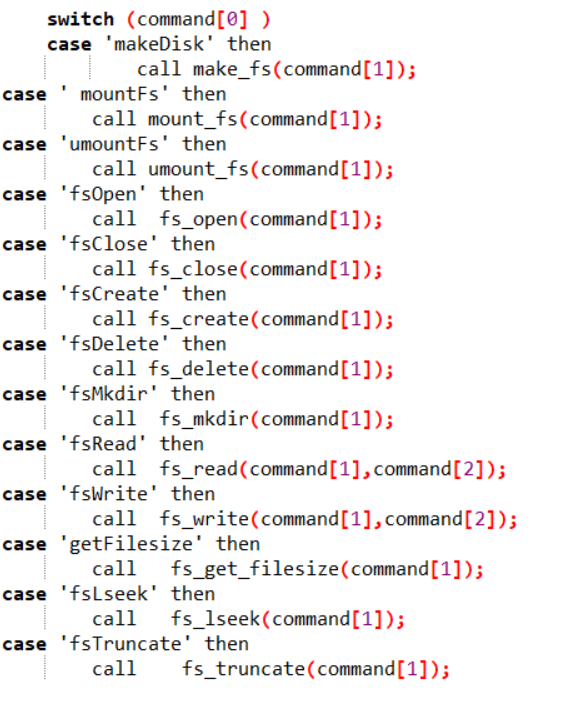
data overflow

Conflict between reading and writing

Memory conflict

Loop cannot jump out

b. Modular testing design

Writing a set of commands from: 

For each input, comparing the change of disk file. If it's the same as expected, it should be correct. Close disk and file, check in the task manager and make sure it’s ensure closed.