The Second Filesystem (SecondFS, UnixV6++ Filesystem)

(For user manual, please refer to "Instructions for use" section directly.)

UnixV6++ filesystem was designed on the basis of UnixV6++, an experimental and educational C++ UNIX V6 implementation under Intel 80386 architecture, written by Chen Hongzhong's team (https://gitee.com/solym/UNIX_V6PP) in Department of Computer Science and Technology, Tongji University.

Introduction

This project is a second-level file system similar to the `Unix` file system, that is, a common large file (`mydisk.img`, called a first-level file) is used to simulate a file volume of the `UNIX V6++` file system.

Lab environment

Operating system：Windows WSL Ubuntu 18.04 LTS

Compiler：gcc-c++ version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)

Compiling

cd to the `Src` directory and execute the `make` command to compile with `g++`. After the compilation is successful, the executable file of `secondfilesystem` will be generated, and the executable file of `./seconfilesystem` will run directly.

Instructions for use

After executing the make operation, the default generated system disk is an empty file, and the unix-style system disk initialization required by the experiment can be completed by the fformat command.

man ： `Help manual`

fformat ： `System initialization`

exit ： `Exit correctly`

mkdir ： `New directory`

cd ： `Change directory`

ls ： `List directories and files`

fcreat ： `Create a new file`

eg:fcreat -r/-w/-rw

fdelete ： `Delete Files`

eg:fdelete

fopen : `Open the file, the fd value of the file will be returned if the file is opened successfully, which is convenient for other functions`

eg:fopen -r/-w/-rw

fclose : `Close file`

eg:fclose

flseek : `Move the read and write pointer, starting from origin, move the read and write pointer to the offset`

eg:flseek

fwrite : `Read content from external file and write to internal file`

eg: fwrite

fread : `Read the content of the internal file and output to the screen`

eg: fread

fin : `To import the host file, you first need to create a file named <filename> in the secondary system`

eg: fin

fout : `Export file to host`

eg: fout

Note: mkdir, cd, fcreat, fdelete, fopen all support relative and absolute paths

---

Testing process

The secondary file system is initially empty, and all folders need to be added manually.

system initialization

`fformat`

`cd home`

Make Sub_directory

`mkdir reports`

`mkdir texts`

`mkdir photos`

File test

`cd texts`

`fcreate test.txt -rw`

`fopen test.txt -rw`

Now assume that the fd is 8, then

`fwrite 8 test.txt 10`

`flseek 8 0 0`

`fread 8 10`

`fclose 8`

System file import and export

`cd /home/photos`

`fcreate temp -rw`

`fin temp icon.png`

`fout temp icon_out.png`

The input and output operations of the experiment report are the same as above

exit

`exit`

After that, perform the corresponding test according to the requirements. For the specific test process, please refer to the full report.