模拟文件系统说明

作者：傅舟涛 瞿凡

一、系统概述

程序有命令行界面和图形界面可选，将用”Filesys.vfs”模拟磁盘，首次运行时会创建磁盘文件并初始化，请注意，**初始化过程中请勿关闭程序**。源码及编译后的程序可从<https://github.com/FanQu/Filesys>获得。

二、名词解释

2.1、路径

路径是用户在磁盘上寻找文件或目录时，所历经的文件线路，用”/”分隔不同的名称。路径分为绝对路径和相对路径。特别的，每个文件夹下都有”.”和”..”两个文件夹，分别表示自身以及它的上级文件夹。根目录的上级文件夹为它自身。

2.1.1相对路径

相对路径，指的是从当前目录开始寻找的路径，格式为”a/b/c”，表示的是当前目录下的a目录下的b目录下的c目录或文件。

2.1.2绝对路径

绝对路径，指的是从根目录开始寻找的路径，格式为”/a/b/c”，表示的是根目录下的a目录下的b目录下的c目录或文件。特别的，**”/”表示根目录**。

2.2、长度

本说明中，长度指的是不包含’\0’的char的个数，或者说是字节数。简单来说，长度指的是有多少个字符（一个中文算两个字符，但本程序禁止使用中文），**如无特殊说明，输入的$path长度不超过1036035。**

2.3、提示

指的是程序输出的内容。提示”>> “表示可以继续输入，除格式化有确认操作提示外，其余若操作成功均无提示，若操作失败会提示失败原因。

2.4、变量与符号$

任何形如”$x”的形式的表示一个变量，通常为输入或输入相关量。常见的有$path（表示文件或目录路径）、$str（表示字符串）、$mode（表示refresh函数的操作种类（见3.5.1））。

2.5、输入限制与使用限制

2.5.1输入限制

输入限制指的是用户输入时，输入的字符串的要求，通常有关于长度和内容的要求，这些要求大多数（除磁盘/目录刷新外）未被程序限制，但不这么做可能出现错误。

2.5.2使用限制

指这样使用时合法的，大多数不合法的操作（删除当前目录或包含当前目录的目录除外）均会返回提示且被禁止操作，注意，虽然程序未限制，但**禁止删除当前目录或包含当前目录的目录**。

三、命令介绍

3.1、路径操作

3.1.1查看当前路径

命令：pwd

内容：查看当前目录路径，输出为绝对路径。

输入限制：无

使用限制：无

实现方式：使用一个双向链表来记录当前工作目录的绝对路径，每次尝试切换目录时复制链表，在寻找目标目录的同时进行链表维护，若目标目录存在则替代当前路径链表，否则释放临时复制的链表。

3.1.2切换目录

命令：cd+空格+$path 示例：cd /user/316

内容：切换至$path的目录。

输入限制：无

使用限制：

1、$path存在，否则会提示”$path No such file or directory”；

2、$path不是文件而是目录，否则会提示”$path is not a directory.”。

实现方式：先复制一条绝对路径的链表，在寻找$path的函数中识别是否是cd调用，若是则同时维护链表。寻找$path的函数采取递归写法，不断地找子目录。

3.2、目录或文件操作

3.2.1查看目录或文件

命令1：ls+空格+$path 示例：ls /user/316

命令2：ls

内容：查看$path的目录或文件的内容。对于目录，显示该目录下的所有文件和目录（不包括”.”和”..”）。对于文件，显示文件名。”ls”等价于”ls .”。

输入限制：无

使用限制：

1、$path存在，否则会提示”$path No such file or directory”；

2、$path不是文件而是目录，否则会提示”$path is not a directory.”。

实现方式：根据记录的路径得到当前工作目录的inode\_id，扫描磁盘文件的对应的目录项，列出除了’.’和’..’的文件夹和文件。

3.2.2重命名目录或文件

命令：mv+空格+$path + 空格 +$new\_name 示例1： mv a.cpp b.cpp 示例2: mv /home/a.cpp b.cpp

内容：将path路径上的文件或文件夹重命名为$new\_name

输入限制： 长度不超过251，只能由大小写字母、数字、小数点"."和下划线"\_"组成，不得包含空格、中文字符和其他符号。

使用限制：

1、$new\_name不能为空，否则提示"The name cannot be empty!"；

2、$path存在，否则会提示"$path No such file or directory"；

3、$new\_name的长度小于251，否则会提示"The Name should be no longer than 251 chars!"；

4、$path所在文件夹不能存在于new\_name相同的文件，否则会提示"$new\_name already existed."

5、$path不能是当前工作目录的绝对路径上的某一级文件夹，否则会出错。

3.3、目录操作

3.3.1创建目录

命令：mkdir+空格+$path 示例：mkdir /user/316

$path格式： “$father\_path/$dir\_name” 或 “$dir\_name”，后者等价于”./$dir\_name”。其中$father\_path表示被创造目录的上级文件夹的路径，$dir\_name表示目录的名称。

内容：创造$path的目录。

输入限制：$dir\_name长度不超过251，只能由大小写字母、数字、小数点”.”和下划线”\_”组成，不得包含空格、中文字符和其他符号。

使用限制：

1、$father\_path存在，否则会提示”$father\_path No such file or directory”；

2、$path不是文件而是目录，否则会提示”$father\_path is not a directory.”；

3、$dir\_name的长度小于251，否则会提示"The Name should be no longer than 251 chars!"；

4、$path尚未存在，否则会提示”$path is already existed."；

5、$father\_path未满，否则会提示"The Directory is full!"。

6、inode（索引）块不能全部被占用，否则会提示”All inodes are used”；

7、block（数据）块不能全部被占用，否则会提示”All blocks are used”。

实现方式：

3.3.2删除文件夹

命令1：rmdir+空格+$path 示例：rmdir /user/316

命令2（**禁止使用**）：rmdir

内容：删除$path的目录以及目录内的所有内容，命令2等价于”rmdir .”。

输入限制：无

使用限制：

1、$path存在，否则会提示”$path No such file or directory”；

2、$path不是文件而是目录，否则不删除且会提示”$path is not a directory.”；

3、为安全考虑，不允许删除根目录，$path为根目录时，不删除且会提示”rmdir: it is dangerous to operate recursively on "$path"”；

4、出于操作安全，**禁止用户删除当前目录，或包含当前目录的目录**，但并未从程序上禁止。当前目录被删除后，当前文件夹位置并不会自动切换，认为当前目录为被删除目录被删除前的地址，使用其他命令可能会造成很多混乱或错误。若用户错误操作导致当前目录被删除，请立刻使用”cd /”将当前目录切换回根目录。

实现方式：依次删除所有它的（除”.”、”..”外）所有子文件夹（递推）或文件，然后删除上级文件夹中关于它的目录项信息，再删除它的”.”、”..”、 superblock信息、inode信息。

3.4、文件操作

3.4.1将字符串写入文件

命令：echo+空格+$str+空格+$path 示例：echo helloworld! /user/316/file

$path格式： “father\_path/file\_name” 或 “file\_name”。其中$father\_path表示被创造目录的上级文件夹的路径，$file\_name表示文件的名称。

内容：将$str写入$path文件，若文件存在则覆盖原文件，否则创建文件。

输入限制：$file\_name长度不超过251，只能由大小写字母、数字、小数点”.”和下划线”\_”组成，不得包含空格、中文字符和其他符号。$str长度不超过4096，且不包含空格，不建议加入中文，可能无法正常读取。

使用限制：

1、$path存在时不能为目录，若$path为目录，会提示”$father\_path is not a file.”；

2、$file\_name的长度小于251，否则会提示"The Name should be no longer than 251 chars!"；

3、$father\_path存在，否则会提示”$father\_path No such file or directory”；

4、$father\_path不是文件而是目录，否则会提示”$father\_path is not a directory.”；

5、$father\_path未满，否则会提示"The Directory is full!"；

6、inode（索引）块不能全部被占用，否则会提示”All inodes are used”；

7、block（数据）块不能全部被占用，否则会提示”All blocks are used”。

实现方式：寻找path路径对应的inode\_id，若未找到，找father\_path的inode\_id，找到后若为目录，寻找新的位置作为文件的地址并创建信息。然后找对应的block\_id进行写入（不满4096用’\0’扩充到4096）。

3.4.2读取文件内容

命令：cat+空格+$path 示例：cat /user/316/file

内容：读取$path文件内容。

输入限制：无。

使用限制：

1、$path存在，否则会提示”$path No such file or directory”；

2、$path不是目录而是文件，否则会提示”$path is not a file.”。

实现方式：判断合法性，找到对应的block，在末尾增加一个’\0’变为4097个char（防止文件内容长4096没有’\0’），然后cout。

3.4.2删除文件

命令：rm+空格+$path 示例：rm /user/316/file

内容：删除$path文件。

输入限制：无。

使用限制：

1、$path存在，否则会提示”$path No such file or directory”；

2、$path不是目录而是文件，否则会提示”$path is not a file.”。

实现方式：删除它的block信息，然后删除上级文件夹中关于它的目录项信息，再删除superblock信息、inode信息、。

3.5、磁盘操作

3.5.1格式化磁盘

命令：format

内容：格式化磁盘，将磁盘中除根目录外的所有内容清空。

使用说明：输入命令后，会提示确认，输入Y确认，输入其他取消格式化；之后会选择模式，输入Q选择快速模式，输入C选择全面模式，输入其他选择取消格式化。

快速模式原理：在不删除根目录的情况下依次删除根目录下的所有内容，相当于没有删除自身信息的rmdir。

全面模式原理：执行创造Filesys.vfs的程序。

3.5.2磁盘/目录刷新

命令：refresh $mode

内容：将程序中记录的内容全部写入文件（$mode为0），或重新读取文件内容（$mode为1）。

输入限制：$mode为0或1，否则会提示"refresh: mode not found, 0 or 1 only"。

使用限制：Filesys.vfs存在，否则会提示” Filesys.vfs is not exist”。

使用说明：程序中不储存block（数据）信息，”refresh 0”只更新inode（索引）和superblock（超级块）信息。正常情况下程序与文件的内容是同步变化的，当文件被替换时会不同步，需要refresh，注意，若用不正常的同名文件替换，很有可能导致程序崩溃。

四、图形界面

4.1、 主界面

主界面是简洁的模拟文件管理系统的图形界面，界面风格基于操作系统。

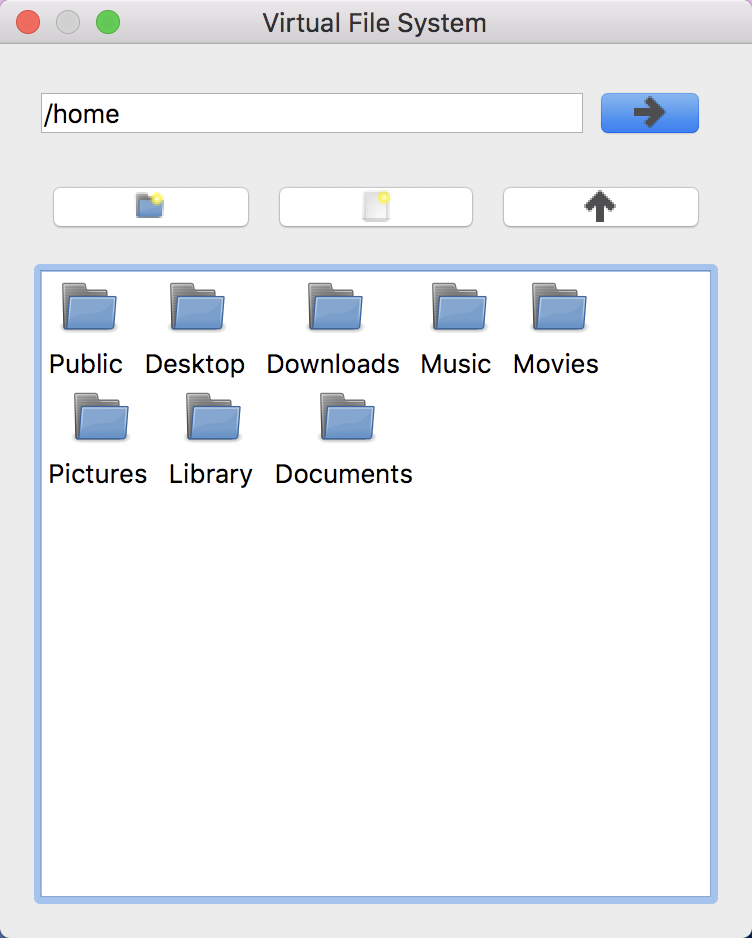
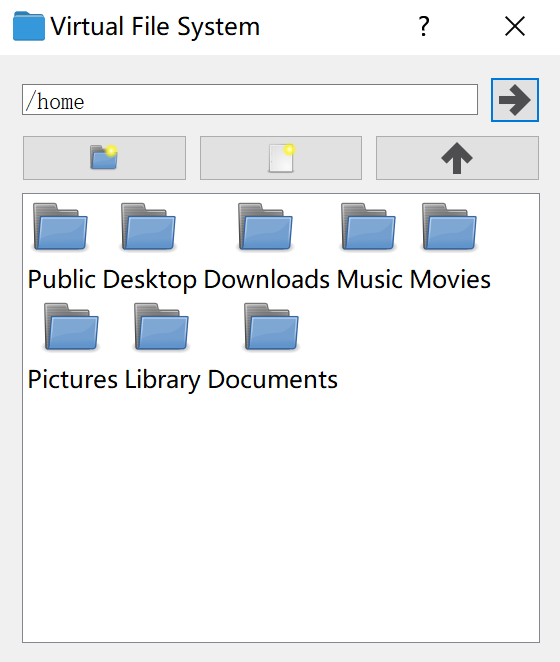
 

图 1

如图是macOS上的界面（图1左）和win10上的界面（图1右）。由标题栏，导航栏，转到按钮，新建文件夹按钮，新建文件按钮，上一级菜单按钮和文件浏览区组成。

4.2、 功能

4.2.1 创建文件夹及文件

创建文件夹和文件两种途径，一种是点击主界面的按钮，一种如图2所示，在空白处选择右键菜单中的新建文件或文件夹。

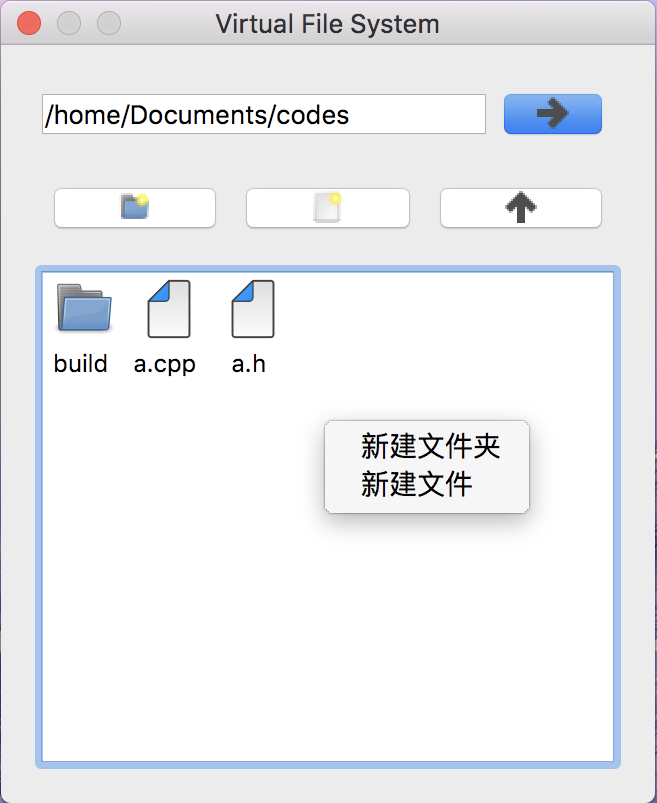


图 2

4.2.2 切换目录

前往某一路径有两种方式，一种是通过双击文件夹进入下级目录和点击上级目录按钮进入上级目录来一层一层前往，另一种是在导航栏输入绝对路径，点击转到按钮来进入。

4.2.3 重命名和删除

重命名和删除的操作方法都是右键文件或文件夹，选择删除或重命名，如图3所示。

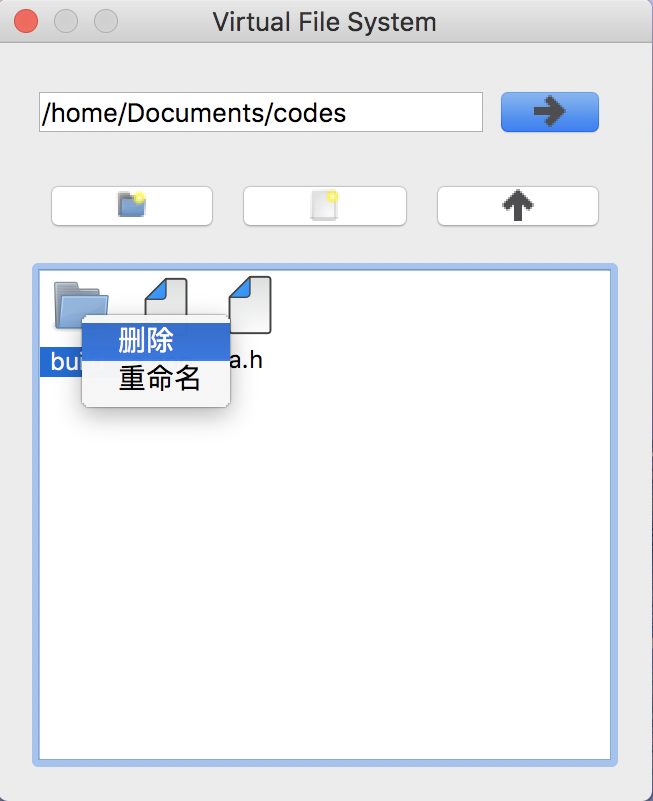


图 3

4.2.4 编辑文件

编辑某一文件可以双击该文件或右键菜单选择编辑，然后会弹出一个简单的文本编辑器界面，主界面有退出、保存、撤销按钮，右键菜单有撤销，剪切，复制，粘贴，删除，全选等选项。如图4所示。另外在未保存即退出时会有保存提醒,如图5所示。

图 4



图 5

4.3、使用限制

基本限制和命令行状况相同，有文件名限制，文件大小限制，文件夹数目限制，对于限制操作有限制提醒。数例如图6所示。

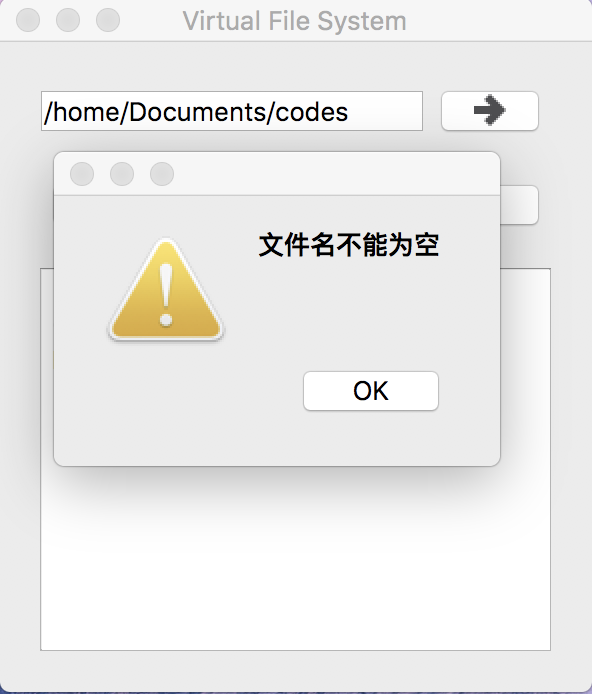
 

图 6

4.4、实现方法

数据维护、与磁盘文件的交互调用原来的底层函数。GUI部分使用Qt编写，用到了，所用的三个主要类继承了QDialog（主界面和文本编辑器）和QListWidget（文件浏览区）。

五、小组分工

傅舟涛主要负责磁盘格式化和磁盘刷新的接口、rmdir、echo、cat、rm、磁盘格式化、以及说明文件的大部分内容，瞿凡主要负责初始化、前端（Gui）、大部分接口、pwd、cd、mkdir、ls、重命名、以及说明文件的重命名和Gui部分。fu.cpp、fu.h的内容来自于傅舟涛，其余主要来自于瞿凡。分工过程中使用github作为远程代码库。