

运行说明

提供了两个运行文件，您可以分别在 Linux Ubuntu 和 win10 下运行。

0 注意事项

特别注意，您需要使用 `exit` 使程序正常退出，否则直接 `CTRL+C` 或者直接中断可能会导致镜像的异常。

同时，`read` 输出到窗口是通过 `cmd type` 进行操作的；如果您的环境不支持，请选择输出到文件吧。

1 镜像的创建与加载

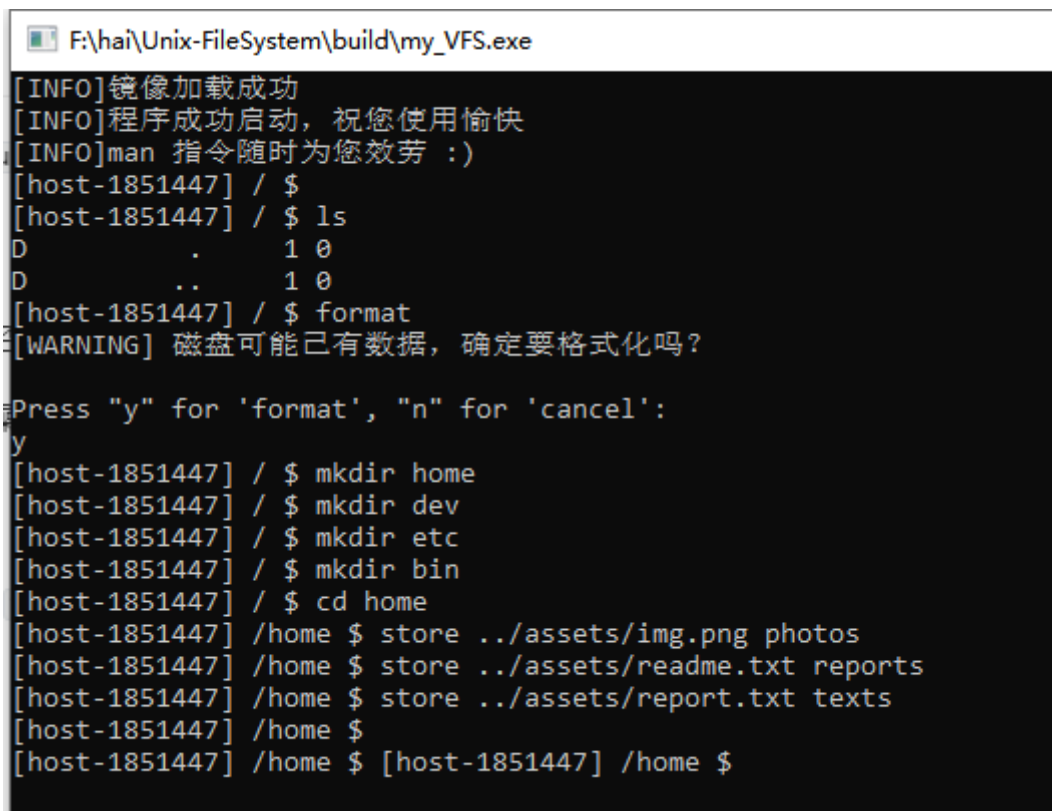
直接运行 `./my_VFS` 即可！如果首次运行（`img` 未产生）会出现提示你是新建了镜像：

```
truth@lion:~/Desktop/hai/Unix-FileSystem/build$ ./my_VFS
[INFO]您刚刚创建了一块新的镜像！
[INFO]成功初始化
[INFO]程序成功启动，祝您使用愉快
[INFO]man 指令随时为您效劳 :)
[host-1851447] / $ ls
D      .      1 0
D      ..     1 0
```

否则就是加载以前的镜像：

```
truth@lion:~/Desktop/hai/Unix-FileSystem/build$ ./my_VFS
[INFO]镜像加载成功
[INFO]程序成功启动，祝您使用愉快
[INFO]man 指令随时为您效劳 :)
[host-1851447] / $
```

你可以重新删除 `1851447.img` 以得到一个全空的镜像；或者可以通过 `format` 以得到一个满足实验报告中 4.1 主程序测试 测试环境的镜像！



```
F:\hai\Unix-FileSystem\build\my_VFS.exe
[INFO]镜像加载成功
[INFO]程序成功启动，祝您使用愉快
[INFO]man 指令随时为您效劳 :)
[host-1851447] / $
[host-1851447] / $ ls
D      .      1 0
D      ..     1 0
[host-1851447] / $ format
[WARNING] 磁盘可能已有数据，确定要格式化吗？

Press "y" for 'format', "n" for 'cancel':
y
[host-1851447] / $ mkdir home
[host-1851447] / $ mkdir dev
[host-1851447] / $ mkdir etc
[host-1851447] / $ mkdir bin
[host-1851447] / $ cd home
[host-1851447] /home $ store ../assets/img.png photos
[host-1851447] /home $ store ../assets/readme.txt reports
[host-1851447] /home $ store ../assets/report.txt texts
[host-1851447] /home $
[host-1851447] /home $ [host-1851447] /home $
```

2 运行提示

每次指令执行都会有提示语句 [INFO] 说明正常运行了;

```
[host-1851447] /test $ fread 0 -o tests 500
[INFO] 成功读出 300 bytes,内容存在 'tests' 中
```

对提示框进行解释:

```
[host-1851447] /home $ [host-1851447] /home $ ls
D      .      101
D      ..     1 0
F      photos 97 15018
F      reports 96 26423
F      texts  95 1545921
[host-1851447] /home $
```

用户名 当前路径

对 ls 进行特别说明:

```

F      texts  95 1545921
[host-1851447] /home $ ls
D      .      101 0
D      ..     1 0
F      photos 97 15018
F      reports 96 26423
F      texts  95 1545921
[host-1851447] /home $
[host-1851447] /home $
[host-1851447] /home $
```

文件大小, 字节数

文件类型 文件名 文件inode节点号

[ERROR] 说明指令操作有问题;

```
[host-1851447] / $ fread 1 -f
[ERROR]不存在这个fd
```

3 指令查询

你可以及时通过 `man` 进行相应的查询! 一共开放了这些接口

```
[host-1851447] / $ man
当前指令      : man
功能简介      : 求助那个男人
使用方法      : man [指令]
可用参数      : 可用[指令] 如下：
                format      : 格式化镜像
                exit        : 正确退出
                touch       : 新建文件（有读写权限）
                mkdir       : 新建目录
                rm          : 删除文件
                rmdir       : 删除路径
                cd          : 进入目录
                ls          : 列出目录及文件的详细信息
                fcreat      : 新建文件并打开文件
                fopen       : 打开文件
                fread       : 从文件中读出
                fwrite      : 写入到文件中
                fseek       : 移动当前指针位置
                fclose      : 关闭文件
                store       : 从外部写入整个文件
                load        : 读出整个文件到外部
[你可以这样使用]: man format
```

比如我忘记了 `fread` 的使用方法，我就可以进行如下查询：

```
[host-1851447] / $ man fread
当前指令      : fread
功能简介      : 从一个已经打开的文件中读取。可以直接读取并打印到终端，也可以指定使输出到外部的目标文件。
                如果是读出并保存到外部文件，可以指定读出的大小或是全部读出。
                注意每一次读出都会修改文件指针的位置！！
使用方法      : fread <fd> <读出大小>
                fread <fd> -f
                fread <fd> -o <外部文件名> <读出大小>
                fread <fd> -o <外部文件名> -f
可用参数      : <fd> fopen返回的文件描述符
                -o <外部文件名> 输出到外部文件
                <读出大小> 指定读取字节数
                -f 全部读出
[你可以这样使用]: fread 1 100
                fread 1 -f
                fread 1 -o FileName 100
                fread 1 -o FileName -f
当出现非法的、或不存在的外部文件、错误的操作权限时，会产生错误提示
[host-1851447] / $
```

4 所有命令一览

```
1
2     static std::string m_man =
3         "当前指令      : man  \n"
4         "功能简介      : 求助那个男人  \n"
5         "使用方法      : man [指令]  \n"
6         "可用参数      : 可用[指令] 如下:  \n"
7         "                format      : 格式化镜像  \n"
8         "                exit        : 正确退出  \n"
9         "                touch       : 新建文件（有读写权限）  \n"
10        "                mkdir       : 新建目录  \n"
11        "                rm          : 删除文件  \n"
12        "                rmdir       : 删除路径  \n"
13        "                cd          : 进入目录  \n"
14        "                ls          : 列出目录及文件的详细信息  \n"
15        "                fcreat      : 新建文件并打开文件  \n"
16        "                fopen       : 打开文件  \n"
17        "                fread       : 从文件中读出  \n"
```

```

18         "                fwrite      : 写入到文件中 \n"
19         "                fseek       : 移动当前指针位置 \n"
20         "                fclose      : 关闭文件 \n"
21         "                store        : 从外部写入整个文件 \n"
22         "                load         : 读出整个文件到外部 \n"
23
24         "[你可以这样使用]: man format \n"
25         ;
26
27     static std::string m_format =
28         "当前指令      : format \n"
29         "功能简介      : 清空镜像文件并恢复到初始状态。 \n"
30         "使用方法      : format \n"
31         "可用参数      : 无 \n"
32         "[你可以这样使用]: format \n"
33         ;
34
35     static std::string m_exit =
36         "当前指令      : exit \n"
37         "功能简介      : 退出时, 保存现场信息。由于有缓存队列的原因, 直接关闭
或'CTRL+C'强行断电会导致缓存内容丢失; \n"
38         "                如果您希望退出并保存这时的镜像的内容, 请使用exit进行安全退
出。 \n"
39         "使用方法      : exit \n"
40         "可用参数      : 无 \n"
41         "[你可以这样使用]: exit \n"
42         ;
43
44     static std::string m_mkdir =
45         "当前指令      : mkdir\n"
46         "功能简介      : 新建一个目录。 \n"
47         "使用方法      : mkdir <目录名> \n"
48         "可用参数      : <目录名> 可以是相对路径, 也可以是绝对路径 \n"
49         "[你可以这样使用]: mkdir dirName \n"
50         "                mkdir ../dirName \n"
51         "                mkdir ../../dirName \n"
52         "                mkdir /dirName \n"
53         "                mkdir /dir1/dirName \n"
54         "当目录已存在、或其父目录不存在时, 会产生错误提示 \n"
55         ;
56
57     static std::string m_ls =
58         "当前指令      : ls \n"
59         "功能简介      : 列出当前目录中包含的文件名或目录名。 \n"
60         "                第一列为文件类型 D 代表目录, F 代表文件 \n"
61         "                第二列为文件名 \n"
62         "                第三列为inode节点号 \n"
63         "                第四列为文件大小, 单位为byte, 目录大小为0。 \n"
64         "使用方法      : <目录名> 可缺省。可以是相对路径, 也可以是绝对路径。默认为当
前目录。 \n"
65         "[你可以这样使用]: ls \n"
66         "                ls dirName \n"
67         "                ls ../dirName \n"
68         "                ls ../../dirName \n"
69         "                ls /dirName \n"
70         "                ls /dir1/dirName \n"
71         "当对象是文件、或目录路径不存在时, 会产生错误提示 \n"
72         ;

```

```

73
74 static std::string m_cd =
75     "当前指令      : cd \n"
76     "功能简介      : 进入目录。 \n"
77     "使用方法      : cd <目录名> \n"
78     "可用参数      : <目录名> 可以是相对路径, 也可以是绝对路径 \n"
79     "[你可以这样使用]: cd ../dirName \n"
80     "                cd ../../dirName \n"
81     "                cd /dirName \n"
82     "                cd /dir1/dirName \n"
83     "当对象是文件、或目录路径不存在时, 会产生错误提示 \n"
84     ;
85
86 static std::string m_fcreat =
87     "当前指令      : fcreat \n"
88     "功能简介      : 新建并打开文件。同时赋予相应的操作权限 \n"
89     "使用方法      : fcreat <文件名> <选项> \n"
90     "可用参数      : <文件名> 当前目录 \n"
91     "                <选项> -r 只读属性 \n"
92     "                <选项> -w 只写属性 \n"
93     "                <选项> -rw == -r -w 读写属性 \n"
94     "[你可以这样使用]: fcreat newFileName -rw \n"
95     "当非法文件名、或非法权限时, 会产生错误提示 \n"
96     ;
97
98 static std::string m_rm =
99     "当前指令      : rm \n"
100    "功能简介      : 删除一个文件。 \n"
101    "使用方法      : rm <文件名> \n"
102    "可用参数      : <文件名> 可以包含相对路径或绝对路径 \n"
103    "[你可以这样使用]: rm fileName \n"
104    "                rm ../fileName \n"
105    "                rm ../../fileName \n"
106    "                rm /fileName \n"
107    "                rm /dir1/fileName \n"
108    "当非法文件名、目录路径不存在时, 会产生错误提示 \n"
109    ;
110
111 static std::string m_fopen =
112     "当前指令      : fopen \n"
113     "功能简介      : 打开一个文件。若要进行文件的读写必须先open。 \n"
114     "                成功打开后, 会返回相应的文件描述符fd。 \n"
115     "使用方法      : fopen <文件名> <选项> \n"
116     "可用参数      : <文件名> 当前目录 \n"
117     "                <选项> -r 只读属性 \n"
118     "                <选项> -w 只写属性 \n"
119     "                <选项> -rw == -r -w 读写属性 \n"
120     "[你可以这样使用]: fopen fileName -r \n"
121     "当非法文件名过长、或目录路径不存在, 会产生错误提示 \n"
122     ;
123
124 static std::string m_fclose =
125     "当前指令      : fclose \n"
126     "功能简介      : 关闭一个文件。读写完毕后, 可以对已经打开的文件进行关闭。
\n"
127     "使用方法      : fclose <fd> \n"
128     "可用参数      : <fd> 文件描述符 \n"
129     "[你可以这样使用]: fclose 1 \n"

```

```

130         "当出现非法的、或不存在的文件描述符时，会产生错误提示 \n"
131     ;
132
133     static std::string m_fseek =
134         "当前指令      :  fseek \n"
135         "功能简介      :  修改当前打开文件的指针位置。可以进行前后移动 \n"
136         "使用方法      :  fseek <fd> <offset> \n"
137         "可用参数      :  <fd> fopen返回的文件描述符 \n"
138         "                <offset> 指定从当前位置要进行的偏移量。正数代表后移，负数
代表前移。 \n"
139         "                本操作保证offset之后，也不会超过当前文件的范
围"
140         "[你可以这样使用]:  fseek 1 500\n"
141         "当出现非法的、或不存在的文件描述符时，会产生错误提示 \n"
142     ;
143
144     static std::string m_fwrite =
145         "当前指令      :  fwrite \n"
146         "功能简介      :  写入一个已经打开的文件中。可以写入终端输入值，也可以写入外
部文件。 \n"
147         "                如果是写入外部文件，可以指定写入的大小或是全部写入。 \n"
148         "                注意每一次写入都会修改文件指针的位置!! \n"
149         "使用方法      :  fwrite <fd> <输入字符> \n"
150         "                fwrite <fd> -d <外部文件名> <写入大小> \n"
151         "                fwrite <fd> -d <外部文件名> -f \n"
152         "可用参数      :  <fd> fopen返回的文件描述符 \n"
153         "                <输入字符> 终端输入的字符串 \n"
154         "                -d <外部文件名> 从外部文件写入 \n"
155         "                <写入大小> 指定写入字节数；当大小超过文件大小时，保证写入
仍是文件大小。 \n"
156         "                -f 全部写入"
157         "[你可以这样使用]:  fwrite 1 123asda \n"
158         "                fwrite 1 -d FileName 100 \n"
159         "                fwrite 1 -d FileName -f \n"
160         "当出现非法的、或不存在的外部文件、错误的操作权限时，会产生错误提示 \n"
161     ;
162
163     static std::string m_fread =
164         "当前指令      :  fread \n"
165         "功能简介      :  从一个已经打开的文件中读取。可以直接读取并打印到终端，也可
以指定使输出到外部的目标文件。 \n"
166         "                如果是读出并保存到外部文件，可以指定读出的大小或是全部读
出。 \n"
167         "                注意每一次读出都会修改文件指针的位置!! \n"
168         "使用方法      :  fread <fd> <读出大小> \n"
169         "                fread <fd> -f \n"
170         "                fread <fd> -o <外部文件名> <读出大小>\n"
171         "                fread <fd> -o <外部文件名> -f \n"
172         "可用参数      :  <fd> fopen返回的文件描述符 \n"
173         "                -o <外部文件名> 输出到外部文件 \n"
174         "                <读出大小> 指定读取字节数 \n"
175         "                -f 全部读出 \n"
176
177         "[你可以这样使用]:  fread 1 100 \n"
178         "                fread 1 -f \n"
179         "                fread 1 -o FileName 100 \n"
180         "                fread 1 -o FileName -f \n"
181         "当出现非法的、或不存在的外部文件、错误的操作权限时，会产生错误提示 \n"

```

```

182         ;
183
184     static std::string m_touch =
185         "当前指令      : touch \n"
186         "功能简介      : 创建文件，拥有读写权限。 \n  "
187         "使用方法      : touch <文件名>\n"
188         "可用参数      : <文件名> 在当前目录创建 \n"
189         "[你可以这样使用]: touch a \n"
190         ;
191
192     static std::string m_rmdir =
193         "当前指令      : rmdir \n"
194         "功能简介      : 删除文件夹。递归删除所有内容。 \n"
195         "使用方法      : rmdir <目录名>\n"
196         "可用参数      : <文件名> 在当前目录创建 \n"
197         "[你可以这样使用]: rmdir folder \n"
198         "当操作对象为文件、或不存在的路径时，会产生错误提示 \n"
199         ;
200
201     static std::string m_store =
202         "当前指令      : store \n"
203         "功能简介      : 整个从外部写入文件。 \n  "
204         "使用方法      : store <外部文件名> <内部文件名>\n"
205         "可用参数      : <外部文件名> 需要写入的文件\n"
206         "                <内部文件名> 内部生成的文件\n"
207         "[你可以这样使用]: store ../asset/readme read.me \n"
208         "当操作对象为不存在的文件时，会产生错误提示 \n"
209         ;
210
211     static std::string m_load =
212         "当前指令      : load \n"
213         "功能简介      : 从内部整个文件读出到外部。 \n  "
214         "使用方法      : load <内部文件名> <外部文件名>\n"
215         "可用参数      : <内部文件名> 需要读出的文件\n"
216         "                <外部文件名> 外部生成的文件\n"
217         "[你可以这样使用]: load read.me ../asset/readme \n"
218         "当操作对象为不存在的文件时，会产生错误提示 \n"
219         ;
220

```