Bobby Templin and Keagen Brendle

CptS 360 Final Project Report

EXT2 File System Level 1

Section A. Introduction:

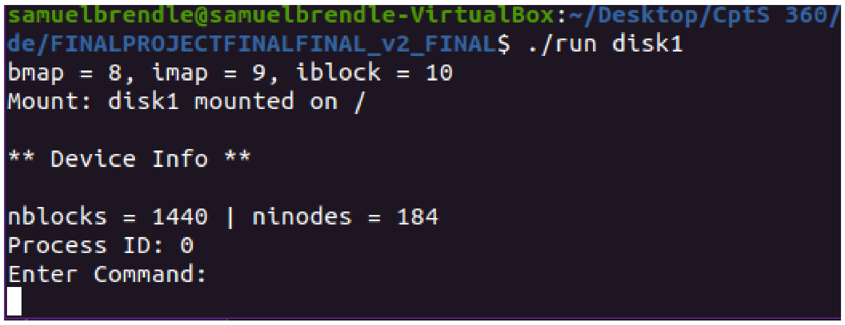
This project was a culmination of everything we have learned in this course so far. Using Linux commands properly, opening/reading virtual disks, and implementing file system functions from provided sample code were all crucial to getting this project to work properly. We were instructed to, at least, implement the functions that would enable use of all level-1 commands for the EXT2 File System in Linux. This included: mount_root, ls, cd, pwd, mkdir, creat, rmdir, link, unlink, symlink, readlink, stat, chmod, utime. The early functions were part of Lab 11, so sample code and virtual disks were used from both Lab 11 and the Final Project files to complete this assignment.

Section B. Design:

For a large majority of this project we adhered to the class textbook, “Systems Programming in Unix/Linux” by K.C. Wang. Chapter 11 goes in depth with the implementation of an EXT2 File System, from the type.h files all the way through Levels 1, 2, and 3. We were provided sample code/starter code in the Lab 11 and Final Project files, and that was used to initially format our code files. From here we went through the FP_CptS360_Spring2021.docx instructions for Level-1 of FS which directed us to create several separate .c files which include: type.h, global.c, util.c, and allocate_deallocate.c as common files. We did a global.h instead and put the allocate_deallocate.c functions in our util.c. Additionally the document outlined: mkdir_creat.c, ls_cd_pwd.c, rmdir.c, link_unlink.c, symlink_readlink.c, stat.c, misc1.c. So we created all of those files, as well as corresponding headers, and made sure to connect our main.c to all of these. We changed up our main so that it operates with a switch statement that accepts input strings and uses those to determine which function to call to execute the command. Initially we had issues with the way our global.h shared global variables with all of the files. There were a lot of “undeclared” and “redefinition” errors, so we changed the way the “externs” were handled with the globals and passed in “char *pathname” as a pointer for the functions rather than a global.

Section C. Results:

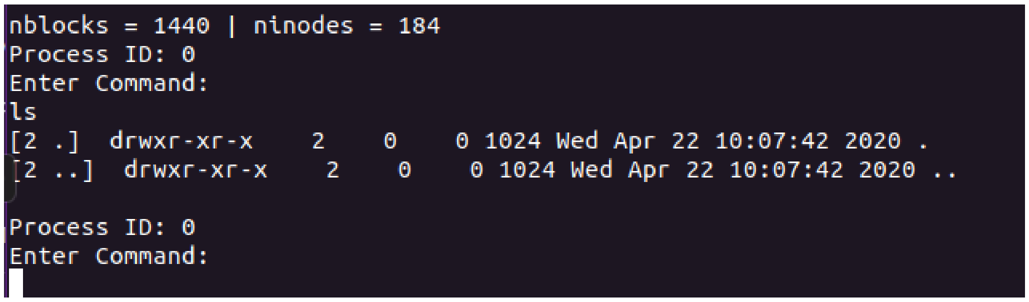
mount_root:



```
samuelbrendle@samuelbrendle-VirtualBox:~/Desktop/CptS 360/
de/FINALPROJECTFINALFINAL_v2_FINAL$ ./run disk1
bmap = 8, imap = 9, iblock = 10
Mount: disk1 mounted on /

** Device Info **

nblocks = 1440 | ninodes = 184
Process ID: 0
Enter Command:
```

Our code after using mount_root for disk1

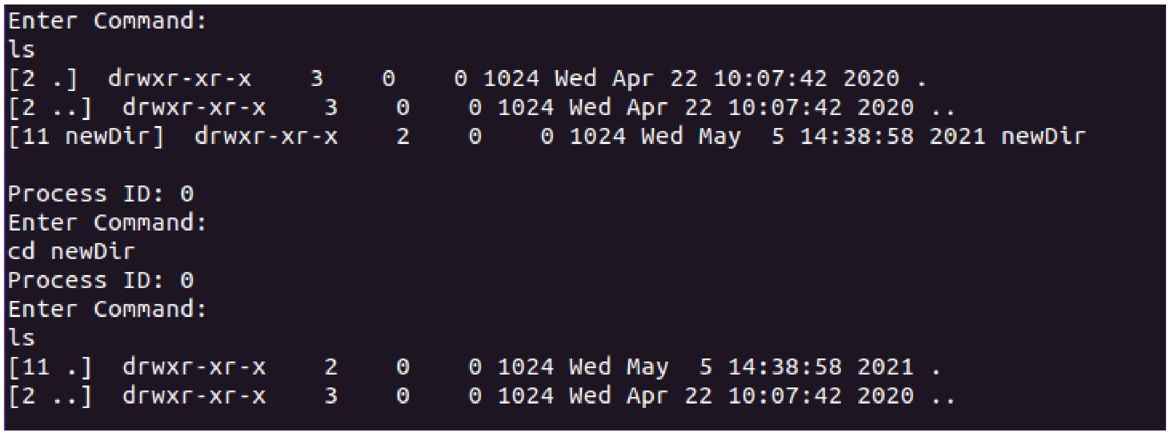
ls:



```
nblocks = 1440 | ninodes = 184
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..

Process ID: 0
Enter Command:
```

The directories from disk1 being listed after calling ls

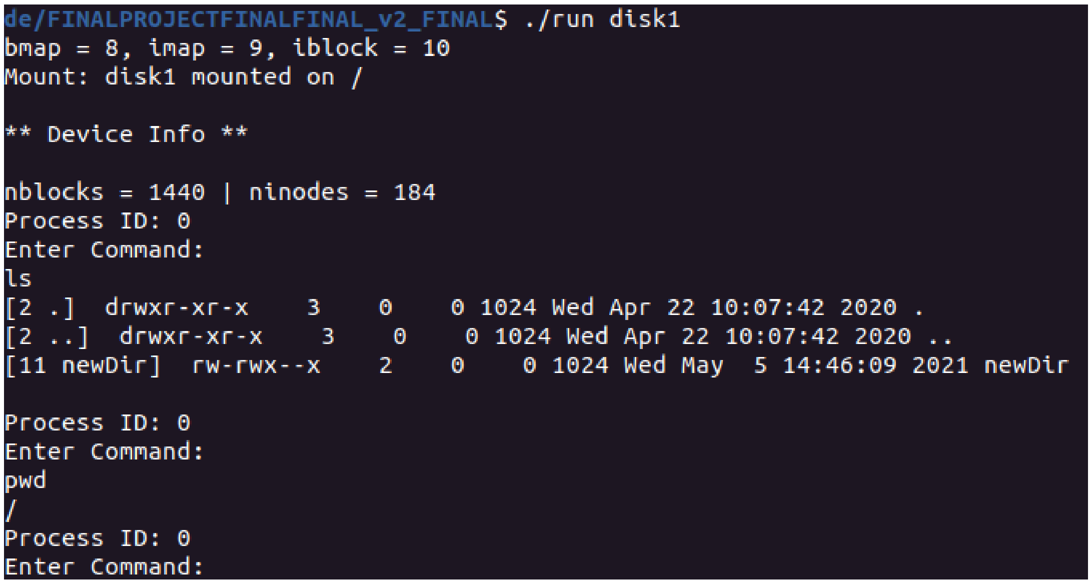
cd:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  drwxr-xr-x    2    0    0 1024 Wed May  5 14:38:58 2021 newDir

Process ID: 0
Enter Command:
cd newDir
Process ID: 0
Enter Command:
ls
[11 .]  drwxr-xr-x    2    0    0 1024 Wed May  5 14:38:58 2021 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
```

Changing directory into newDir and listing its contents

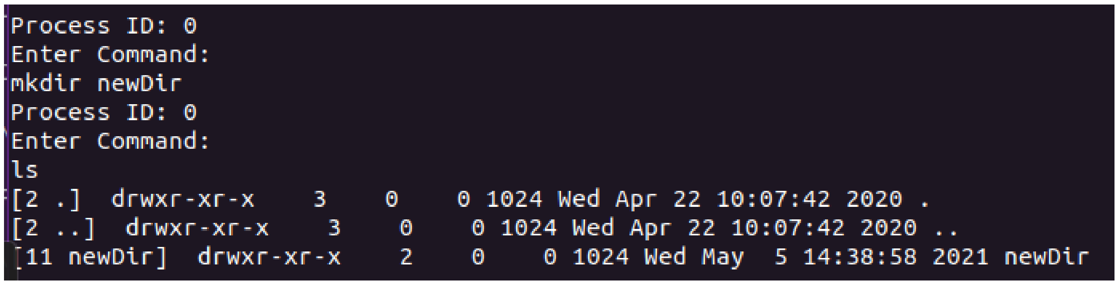
pwd:



```
de/FINALPROJECTFINALFINAL_v2_FINAL$ ./run disk1
bmap = 8, imap = 9, iblock = 10
Mount: disk1 mounted on /

** Device Info **

nblocks = 1440 | ninodes = 184
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  rw-rwx--x    2    0    0 1024 Wed May  5 14:46:09 2021 newDir

Process ID: 0
Enter Command:
pwd
/
Process ID: 0
Enter Command:
```

Calling pwd after loading in disk1, root "/" is printed

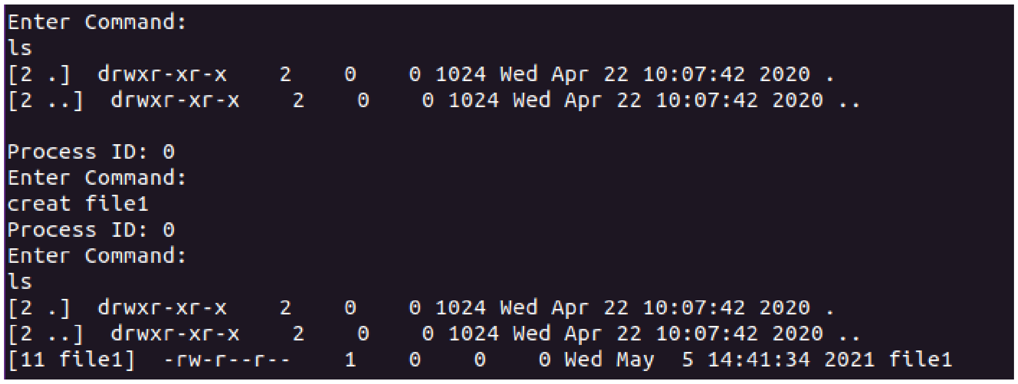
mkdir:



```
Process ID: 0
Enter Command:
mkdir newDir
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  drwxr-xr-x    2    0    0 1024 Wed May  5 14:38:58 2021 newDir
```

Calling mkdir to create newDir, then listing all directories and having it displayed

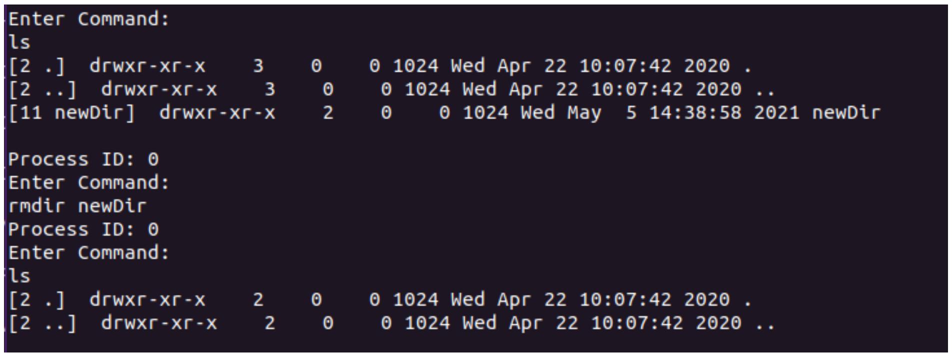
creat:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..

Process ID: 0
Enter Command:
creat file1
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    1    0    0    0 Wed May  5 14:41:34 2021 file1
```

Creating a new file "file1" with creat

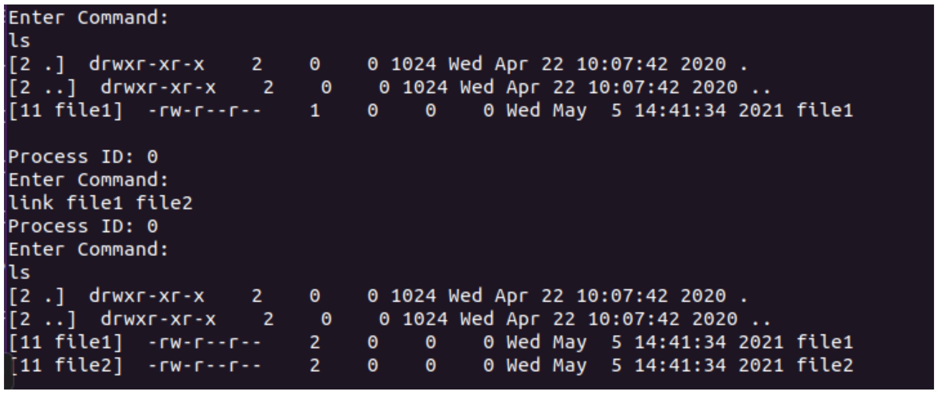
rmdir:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  drwxr-xr-x    2    0    0 1024 Wed May  5 14:38:58 2021 newDir

Process ID: 0
Enter Command:
rmdir newDir
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
```

After having created "newDir", rmdir newDir is called and contents are listed showing that it no longer exists

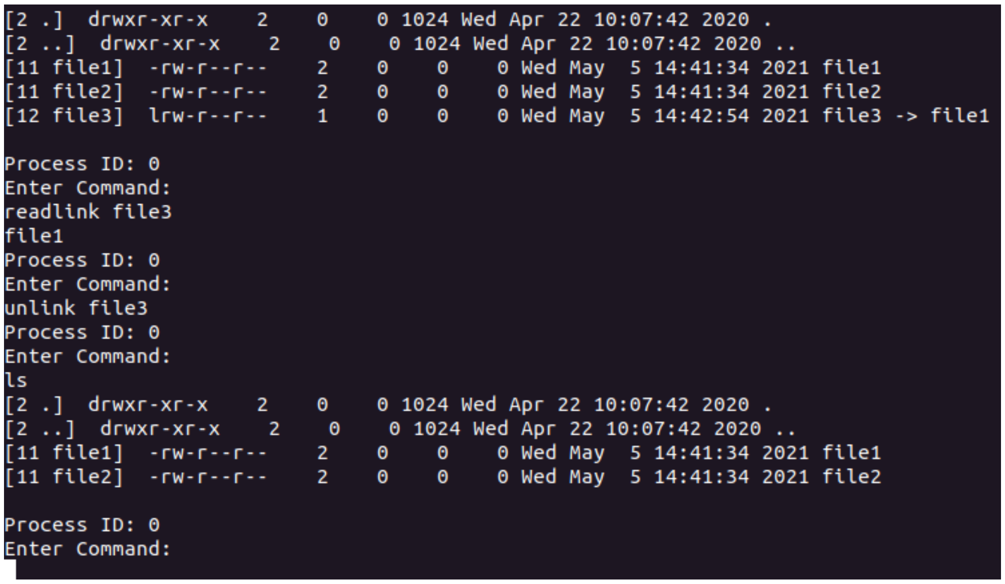
link:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    1    0    0    0 Wed May  5 14:41:34 2021 file1

Process ID: 0
Enter Command:
link file1 file2
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2
```

Calling link to link "file1" to a new file, "file2" and listing all files afterward

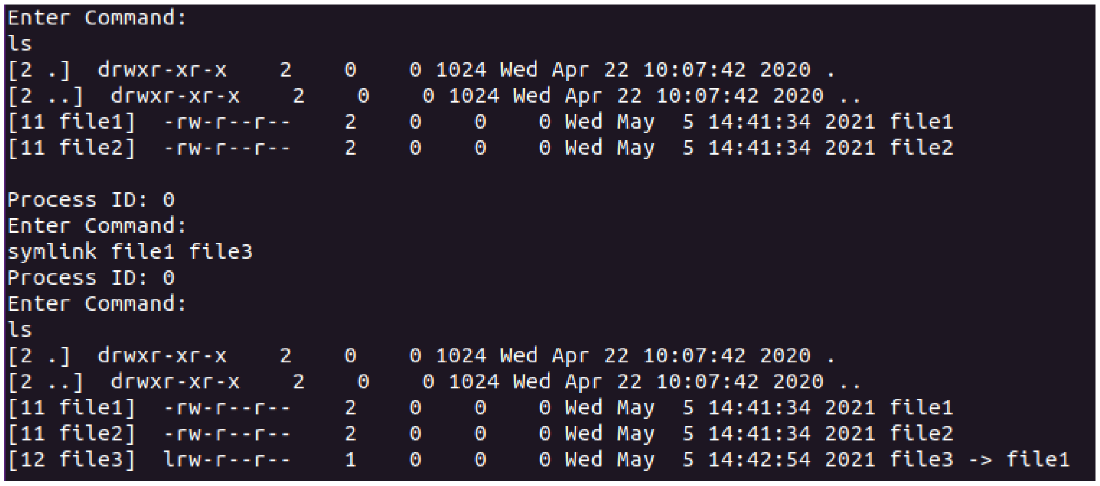
unlink:



```
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2
[12 file3]  lrw-r--r--    1    0    0    0 Wed May  5 14:42:54 2021 file3 -> file1

Process ID: 0
Enter Command:
readlink file3
file1
Process ID: 0
Enter Command:
unlink file3
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2

Process ID: 0
Enter Command:
```

After using "symlink file1 file3" unlink is used to remove file3, and contents are listed showing it no longer exists while "file1" and "file2" still remain

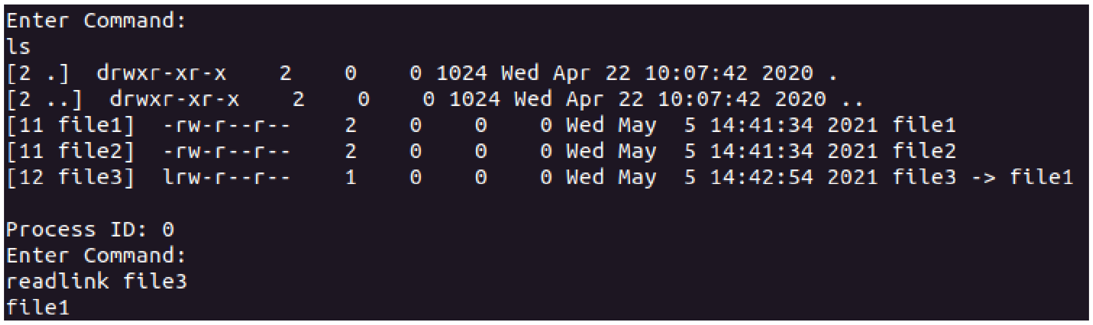
symlink:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2

Process ID: 0
Enter Command:
symlink file1 file3
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2
[12 file3]  lrw-r--r--    1    0    0    0 Wed May  5 14:42:54 2021 file3 -> file1
```

"file1" and "file2" have already been created, and symlink is used to slink file1 to a new "file3". After displaying the contents it can be seen that file3 has an LNK type and is pointing to "file1"

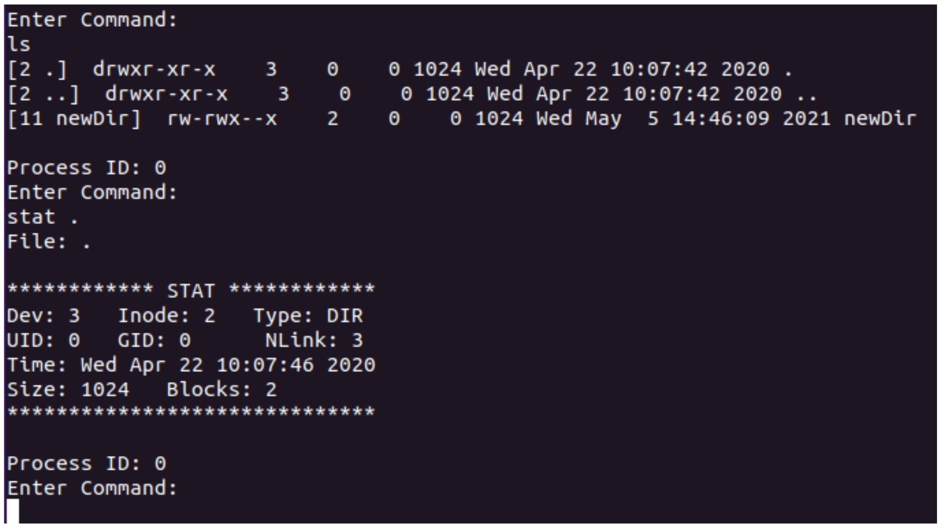
readlink:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    2    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 file1]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file1
[11 file2]  -rw-r--r--    2    0    0    0 Wed May  5 14:41:34 2021 file2
[12 file3]  lrw-r--r--    1    0    0    0 Wed May  5 14:42:54 2021 file3 -> file1

Process ID: 0
Enter Command:
readlink file3
file1
```

Read link is used on file3, which had been symlinked from file1

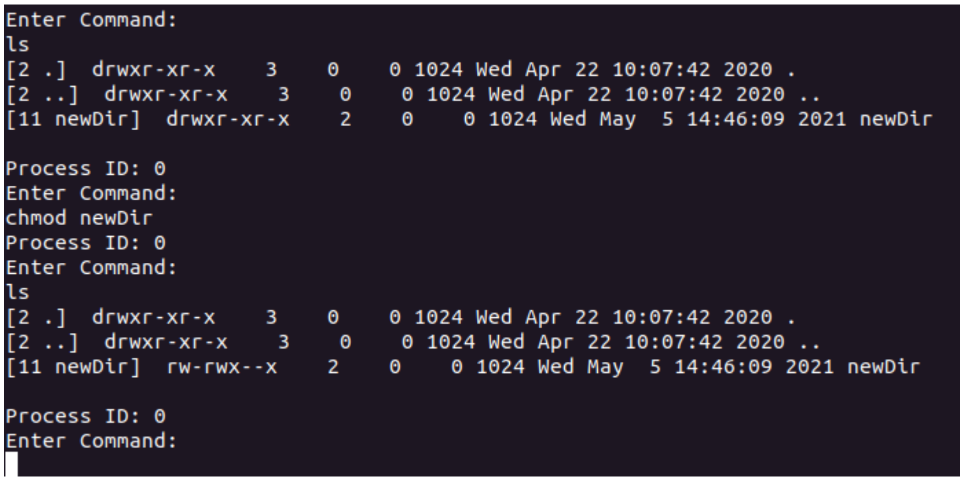
stat:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  rw-rwx--x    2    0    0 1024 Wed May  5 14:46:09 2021 newDir

Process ID: 0
Enter Command:
stat .
File: .

************ STAT ************
Dev: 3   Inode: 2   Type: DIR
UID: 0   GID: 0      NLink: 3
Time: Wed Apr 22 10:07:46 2020
Size: 1024   Blocks: 2
******************************

Process ID: 0
Enter Command:
```

stat is used on the first directory "." within disk1

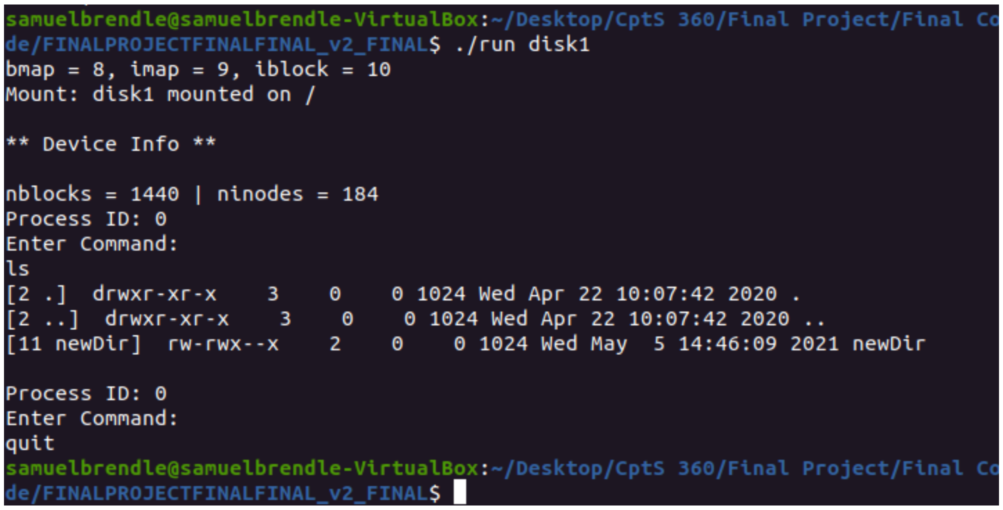
chmod:



```
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  drwxr-xr-x    2    0    0 1024 Wed May  5 14:46:09 2021 newDir

Process ID: 0
Enter Command:
chmod newDir
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  rw-rwx--x    2    0    0 1024 Wed May  5 14:46:09 2021 newDir

Process ID: 0
Enter Command:
```

chmod is used on "newDir" and the permissions change from "drwxr-xr-x" to "rw-rwx—x"

quit:



```
samuelbrendle@samuelbrendle-VirtualBox:~/Desktop/CptS 360/Final Project/Final Co
de/FINALPROJECTFINALFINAL_v2_FINAL$ ./run disk1
bmap = 8, imap = 9, iblock = 10
Mount: disk1 mounted on /

** Device Info **

nblocks = 1440 | ninodes = 184
Process ID: 0
Enter Command:
ls
[2 .]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 .
[2 ..]  drwxr-xr-x    3    0    0 1024 Wed Apr 22 10:07:42 2020 ..
[11 newDir]  rw-rwx--x    2    0    0 1024 Wed May  5 14:46:09 2021 newDir

Process ID: 0
Enter Command:
quit
samuelbrendle@samuelbrendle-VirtualBox:~/Desktop/CptS 360/Final Project/Final Co
de/FINALPROJECTFINALFINAL_v2_FINAL$
```

quit is called and the program successfully closes and returns me to the normal terminal

Section D. Conclusions:

Overall, this project was a great way to warp up this class. We mention this in almost every lab report, but these assignments have greatly improved my comfort/experience with Linux. This project especially helped me with my understanding of a few commands I had never used before, so knowing how to properly use them all in Linux was helpful before beginning to implement them all ourselves. Issues were had with the structure of our project, as initially there are variables that are provided as globals and some kind of “extern” reference is needed to actually use them in other files (at least initially). We began formatting all of the .c and .h files, and from there passing in these variables created “redefinition” and “undeclared” variable errors. To fix this we made it so that the functions in other files accept char pathname as a pointer instead. This was already mentioned, but we also made it so that our main has a switch statement that controls what functions are called to execute commands. Here and there we found discrepancies in the code that is outlined in the textbook (typos, contradictions, things that are present in the textbook but not the sample code) so making sure that we adjust for these findings was crucial to getting everything working.