

Aaron Steele

Part 1

- I had the hardest time getting args-none to work. Professor Harrison helped a lot on Canvas. Once I got the programs actually running the rest was fairly simple, just parsing out the file name and then all the tests passed.

Part 2

SYS_CREATE

- Copied the write. My only problem was figuring out how to talk to the filesystem. The filsys.h cleared this up for me.

SYS_OPEN

- Make a int and put it in file.c. Hopefully that's okay to do. This will be the external file ID.
- We need a global counter, so g_file_num will do that. I'll init it to 3 in syscall_init, which is awful coding practice but whatever.
- I don't know how to link the ID to the actual file pointer yet. I'm thinking an array? I'll save that until reading, I guess.
- I think that moving the file struct into file.h will fix syscall.c not knowing it exists? Yes, that fixed it.
- Also apparently booleans aren't in C by default? So including stdbool.h was required.

SYS_READ

- I couldn't figure out how to get the filesystem to read a file, but then I checked file.h and saw that's where the reading code was for some reason.
- Now I need to convert the fd into the file pointer. I don't save that in open yet, so I need to go back and do that too. But how.
- I implemented an array that's awful practice but whatever.
- Now apparently I need to do sys_filesize

SYS_FILESIZE

- Someone on piazza mentioned file_length, so I'm gonna try to use that.
- I moved my code for checking if the file was in the array to a function get_file
- I used file_length, and it appears to work properly. Along with read, too. Now it's yelling at me about syscall 12, which is close. So that's next.

SYS_CLOSE

- I should just be able to zero out the array at the given index and that should properly close it?
- Oh, and use file_close in file.c.

That...actually worked the first time I ran it. I'm surprised.

It's obviously terrible practice because with the straight incrementing the OS can only support 256 total file reads/writes before it refuses to do any more, but I'm running low on time and that seems fine to pass the tests. An actual implementation would probably use a list or something.

`SYS_WRITE`

- I pretty much completely copied the code from read and change `read_file` to `write_file` and everything worked perfectly. So...yay?