

Tyler Crabtree

CS460
Project 1

Part 1:

Essentially, an executed `read()` with no open files will exit with the value `-1`, but I will explain the process in detail below.

Here is a list of steps that start in a user program will call `read(10, buf, n)`:

1. The user program will push `n` bytes.
2. Then, the user program will push a reference to `buf`, (`&buf`).
3. Before calling a system call, the items will be pushed on the stack (push `fd`).
4. The call to `read()` will be evoked.
5. The code for `read` will be put in a register.
6. Execute TRAP of the kernel (library will place arguments in registers to issue this TRAP).
7. The kernel will call the sys call handler.
8. Sys call handler evoked.
9. Read with user program.
10. Return to the caller.
11. Finally, increment the SP.

Part 2:

Converted files to implement sys call.