**Environment:** I’m running and have been working on this project on a CentOS Linux 7.8 system via MobaXterm.

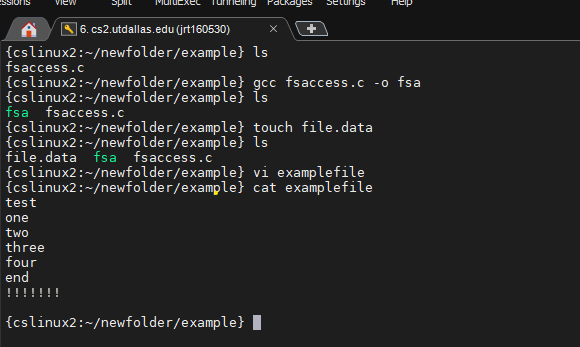
**Purpose**: This document will show an example of how the program is run.

**Background:** This project was written for an Operating System course. There will be many areas of improvement in this program. I had previous experience writing code in C, but many of the concepts, libraries, and methods were new to me and was done within a pretty narrow window of time.

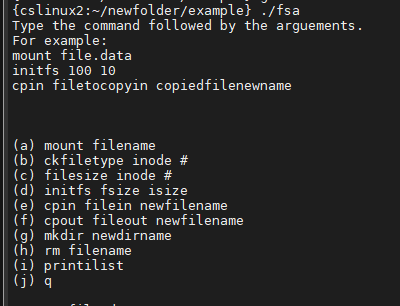
Refer to the “*READ ME.txt*” document for more information.

1. Initially I upload and compile the program fsaccess.c and created the necessary files.

* “file.data” is used as the mounted file system.
* “examplefile” will be used as the file we will copy into the file system and afterwards back out.

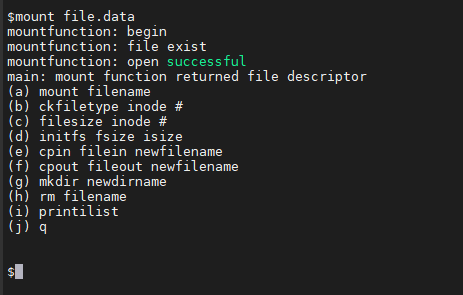


1. Run the compiled program named “fsa”. Upon running the program, the user will be prompted with a list of commands.

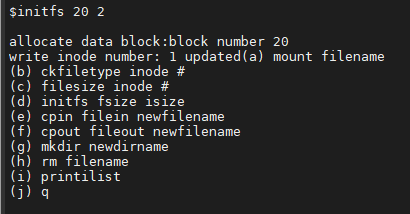


1. The first thing that needs to be done is to mount the file system. In this example, “file.data” is our file system.

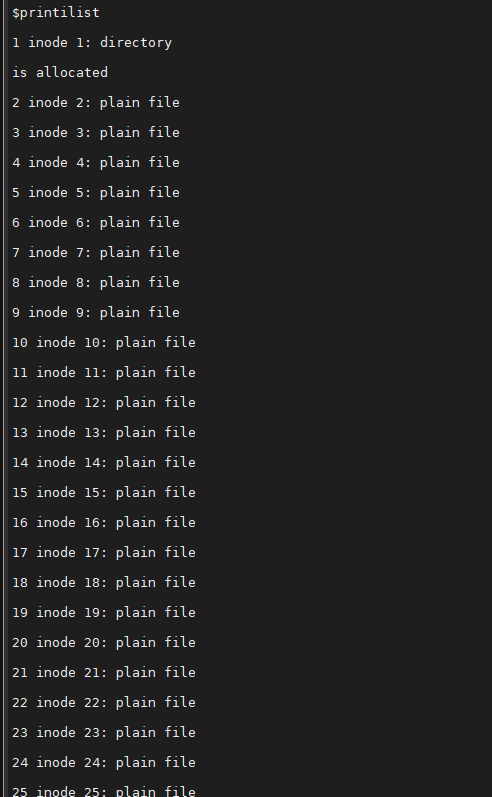
**Note:** throughout the program console messages will be displayed which were used to test the program.



1. After the file system is mounted, initialize the filesystem memory size (fsize) and size of ilist (isize).



1. After the file system is initialized, we can check the ilist with “printilist”, copy files in, copy files out, and quit and reopen the program to the saved file system (after we remount). In the proceeding steps we will show these examples.
2. Printilist shows the ilist. There are 16 inodes per blocks allocated. In the previous initfs command we allocated an isize of 2, so we have a total of 32 inodes. Inode 1 is out superblock, is a directory, and is allocated. All other inodes are not allocated and by default are shown as plain files (but not allocated).



1. Next we will run the copy in command. We can see that inode is now allocated.

