

**// Seongwoo Choi**  
**// scho29**  
**// 1368039**  
**// CMPS 111**  
**// Winter 2017**  
**// Darrell Long**

Purpose:

The purpose of this project is to get set up with freeBSD operating system on our machine and use git repository to submit assignment. The first assignment (or assgn0) was to let students to get used to new operating system.

How I approached the problem:

When I was designing and programming this assignment, I had to understand how concatenate works. According to GNU/Linux organization, a cat program opens the content of file and print the content on standard output. Based on this understanding, I had three thoughts on designing my version concatenate program in C language. First of all, the cat program should understand the command of a user. When a user types './cat', the program should imitate what the user types on the screen. Secondly, a cat program should know how to determine if there is a file or not. What this means is that if a file exists based on its name, a cat program should open the content of that file and then print out the content of that file. Lastly, a cat program should print error messages when there is no file based on its name.

What system calls I will use:

According to GNU organization, 'system call is requests for service that a program makes of the kernel'. System calls I will use are open(), close(), read(), write(), and perror(). The open() function is declared in the header file fnct1.h and read(), write(), and close() functions are declared in unistd.h. This is the reason why I will include those two libraries to start with. errno.h will be needed to print out an error message on the screen.

How you will open files, read then, write stdout, handle errors:

I will first list the integers in the beginning of the cat.c.

int main will take in the values of int argc and argv[]. argc will be the number of strings pointed to by argv[]. Error messages are printed when there is no such file that user has written on screen and when there is no content in a file. The code will return to -1 so that it can quit the process.

Design of the code:

My program will be initiated on the command line with the "./cat" command. I will also specify the files that are being used on the command line as well. So the program will be executed using the "./cat file1 file2 ..." where file1 and file 2 are the names of the files that this program will read and print the contents for. As this program by its very description prints the contents of a file, if there is no file given during the execution of this program, on the command line, then the program will return an error, otherwise, print the contents to stdout. It will also be able to read

multiple files at the same time and print their contents. I will specify the files that I want to be read on the args line on a command line.

So this program, in simple terms, upon execution will check command line to see what files we are talking about and go to those files, read their contents and print them to stdout. If no file is specified or if the file is missing then we will get an error message as specified in the code.