Design Document for Assignment 0

William Kudsk

CRUZID: wkudsk

CSE 130, Fall 2019

1. Goal:

The goal of this assignment is to create a program named dog that function just like the cat linux command.

2. Assumptions:

I am assuming to pass this assignment that cat and dog should print out the exact same content to standard output if given the exact same arguments.
I am also assuming that my program only needs to function as described on ubuntu.

3. Design:

For this program I wrote in C++, using the unistd.h, iostream.h, string.h, err.h, and fcntrl.h header files. These were needed to use certain functions like open(), warn(), write(), read(), and strcmp(). The program iterates through all the arguments in argv[] up to argc. If a dash is found or no arguments are given, the function continuously reads input from standard input and writing it back to standard output, until given an end of file flag from the user. This end of file flag can be given by pressing the control key along with the ‘d’ key on the keyboard. Once that is given the program will continue to read any additional files given to it before terminating. The program reads files by opening them with the open() function, checks for any errors, and then reads them and writes to standard output. The program has to be able to read a file of any size, so the program must loop over the read and write function until read() returns an end of file value of 0. The file is then closed and the program moves onto the next argument in argv.

4. Pseudocode:

Program dog:

For all given arguments in argv

if(dash is found or no arguments) - read from stdin and write to stdout until ctrl+D is given

Else

open(file in argv[i])

If(file cant be opened or file is a directory) warn(file)

Else

Int eof = any non-zero number

While you haven't reached end of file:

buffer[32768] = ""

eof = read(file, buffer) - store into buffer sized 32768

write(stdout, buffer)

close(file)