# Lab 0 Design Document: dog

Darren Dawson
dwdawson@ucsc.edu
CSE 130, Fall 2019

## 1. Goal

The goal of this program is to implement the default functionality of the 'cat' command: to 'concatenate files and print on the standard output'. The program will be called 'dog' and will not need to handle any of the flags that cat can. In the case of errors, dog should print an error message to standard error.

## 2. Limitations / Constraints

For this lab, we are **not** allowed to use any of the C Library FILE * Functions. Basically, if a function takes in a file as an argument, it is off limits. These include:

- fread()
- printf()
- fopen()

The following functions **are allowed** for printing errors:

- fprintf()
- perror()
- warn()
- sprintf()

The following system calls are allowed (and expected) for interacting with files

- open()
- read()
- write()
- close()

## 3. Assumptions

We are assuming this program is being run on ubuntu, the compiler is clang, and files being passed in are in the local directory. This program is not built with arbitrary filepaths in mind.

## 4. Design

The design of dog will be simple: for each argument passed into the program, open the file into a byte buffer and print each byte one at a time.

## 4. Pseudo Code

```
Loop over arguments:

    If (current argument is not a valid filepath)
        Write to std error and skip it

    Otherwise
        While (there are unread bytes left of this file):
            Load the bytes in range (n, n + m) of this file this argument points into a
                buffer
            Print the contents of the buffer
            Increment n by m so we are pointing at unread bits
```