Assignment 0: Dog Design Doc eingeman

1) Goals

The goal of this assignment is to mimic the functionality of the 'cat' in the linux system with my own C++ program called dog. But, my program will not support any flags.

2) Design

Part 1: Function main

1a) First, we must check whether our user inputted less than 2 args. If true, we will just read from STDIN and write it back to the STDOUT and be done.

```
~ if(argc < 2){
        print_func (STDIN, STDOUT);
  }
```

1b) If there are > 2 args, then we must loop through them all in order. This can be done with a simple for loop.

```
  for(int i = 1; i < argc ; ++i);
```

1c) Once in the for loop, we must check if the file name is equal to '-'.

```
    if(*argv[i] == '-'){
       //read from STDIN and write to STDOUT.
    }
```

1d) Then, we can read try to open the file at argv[i] and if it returns -1, we know an error occured and we need to call warn(). Else, we can call our print function and then close the file.

Part 2: print function

1) The print function will take two arguments, the in file descriptor and out file descriptor.
2) The function will the allocate a buffer for reading and writing.
3) While the read is greater than 0, "meaning it has bytes to read, we must write" the same bytes to the stdout. Put this inside a variable.

In code:

```
while (var=(read(infiledescriptor, &buffer, buffer size))>0){
    write(outfiledescriptor, &buffer, var);
  }
```