Assignmet 0 Design Document

Matthew Tan
mxtan
CMPS 111, Fall 2018

1 Goal

The goal of this program is to display the contents of a file or files to standard out. If a file is less than 10 lines, the entire file is printed, otherwise the last 10 lines are printed.

2 Assumptions

I am assuming that newlines are represented by the newline character, so this will not work in Windows or Mac.

In general, the number of lines is equal to the number of newline characters. However, if the last line in the file isn't terminated by a newline, it still counts as a line anyway, which would add one to the total number of lines without a newline.

3 Design

The general approach is to read the file line by line into a char pointer buffer. The lines are stored in a list of char array. If the number of lines is less than 10, the file is simply printed to standard out. If the number of lines is greater than 10, we enter a loop and continuously push off the first entry in the list to make room for the other entries in the file. When the file is finished reading the line list is printed to standard out.

4 Pseudocode

This is the pseudocode for the program.

```
// declare file size and current position

// stringlength function
int stringlength(cost char* s)
    count ← 0
    while s[count] != NULL
        count++      return count

// readline function
```

```
int readline(fd, buffer, len)
    declare a variable t
    for i = 0 to len
        readFd = read(fd, &t, 1)
            if current position < file size
                if t is new line
                    buffer[i] is t
                    buffer[i + 1] is new line
                    readLen bump up by 1
                else
                    buffer[i] = t
                    readLen is i
            else
                buffer[i] is t
                buffer[i + 1] is new line
                readLen bump up by 1
                return -1
close loop
return -1

// main function
first check command line arguments
    for fileIdx to argc
        fd = open(argv[fileIdx], read only)
        if fd is less than 0, print error message
         initialize some variables such as read and write file descriptors, as well
as a buffer and line list (size of 10)
        set line list to null
        using stat.h get the size of file in bytes
        initialize entire readLenList to 0
            while (!done)
                malloc the buffer
                read a byte from the file
                check if lineNo < 10
                    set line list to buffer
                else
                start a loop j = 0 to 9
                    shift line list by 1
                    shift read len list by 1
                    last line is assigned to buffer
                    bump up len of line list
                bump up lineNo
end while
                if byteRead is < 0
                    done = 1
                    break
```

// print out line list accordingly
close fd