CSCI 330 Spring 2017 Assignment 2

Purpose

The purpose of this assignment is to practice reading and writing from files using the UNIX system calls. You will be writing a basic implementation of the `cat` command using C++.

Description

If you recall, the `cat` command takes a list of files as command line arguments, and then opens each file, dumping its contents to standard output, in the order the filenames were passed in.

You will be implementing this behavior.

Requirements

- Your program must handle any number of files, which will have their filenames passed as command line parameters.
- No matter how long each file is, your program must display all of the data inside.
- If an input file is not a text file, make sure that all of the data is displayed. This means that you will not be able to use `cout` on its own to output the data.
- You must use the UNIX system calls we spoke about in class to implement the reading portion. You can use them for the writing portion as well, but this is not required. This means that you cannot use the C++ file stream classes for the file input (`open`, `read`, `close`).
- Make sure to clean up after you are done with each file, calling `close` on its file descriptor.

What to turn in?

All you need to turn in for this assignment is the C++ source code file. It should be submitted via Blackboard, and it should have one of the file extensions that are appropriate for such a file (`.C`, `.cc`, `.cpp`, etc.). Make sure to appropriately document your code. This includes a documentation box at the top of your program, as well as before each function you write, and comments on the code that actually does the work to explain what is going on. If documentation is missing, your grade on the assignment will suffer.