Assignment 4 – XD Report

Jeremy Miller

CSE 13S – Winter 24

Purpose

This program reads a file and prints out the contents, as well as the bytes in hexadecimal. It is a buffered reader that reads 16 bytes at a time.

Questions

Please answer the following questions before you start coding. They will help guide you through the assignment. To make the grader's life easier, please do not remove the questions, and simply put your answers below the text of each question.

- What is a buffer (talk about the data type and the purpose)? Why use one?
- A buffer is a block of memory that stores data temporarily. It is useful in reading files, as you can store the contents of the file temporarily, do something with it, then get rid of the buffer.
- What is the return value of `read()`? What are the inputs?
- Read() is a function in C that reads data from a file into a buffer. The inputs are an integer file descriptor, a pointer to the buffer, and the minimum number of bytes to be read.
- What is a file no. ? What are the file numbers of `stdin`, `stdout`, and `stderr`?
- a file number is an integer that identifies an open file. It is also referred to as a file descriptor, and is one of the inputs of read().
- What are the cases in which `read(0,buffer,16)` will return 16? When will it *not* return 16?
- It will return 16 when the inputs are valid, and there is a sufficient amount of data available. It will not return 16 if fewer than 16 bytes are available and it reaches EOF. Also if any errors occur.
- Give at least 2 (very different) cases in which a file can not be read all at once
- Two examples of situations where a file can not be read all at once are if the file is too large, or if the file being read does not have a set size to begin with. An example of this would be if the file was being updated in real-time, the whole file would not be able to be read at once since new data is coming in constantly.
- What is the range of char? A byte? The ASCII table? Which range should your program accept (if any of those)
- The range of a char depends on whether it is signed or unsigned, if it is signed it is -128 to 127, and unsigned is 0 to 255. The range of a byte is also 0 to 255, as it can represent values 2 to the power of 8. The ASCII table defines 128 characters. My program should accept the ASCII table from 32 to 126 as that covers all English characters.

- What is the decimal equivalent of the 8 bit integer 0b1001 0110? [1]
- The decimal equivalent of this 8 bit integer would vary depending on if it is signed or unsigned. If it is signed, it would be -106. If it is unsigned, it would be 150. It is negative because the most significant bit is 1.
- Convert the 8 bit integer above to a 32 bit integer. [2]
- The 8 bit integer above as a 32 bit integer is: 11111111 11111111 11111111 10010110.
- What does the `%X` format specifier mean? What type of data does it expect? [3]
- The upper case X format specifier is utilized to specify an unsigned integer as a hexadecimal.

Testing

In order to test my program, I will create a a testing script that creates a bunch of different scenarios for my program to tackle. Each of the input files will have different contents, some that is meant to work and some that is meant to break the code in order to make sure my program works in all cases.

How to Use the Program

This program reads a file and prints out the contents, as well as the bytes in hexadecimal. In order to make use of this program, you must supply an argument when you run the program. The single argument given should be the name of the file you want to have read by the program. After running the program with the filename as the only input, the program will work. If there is no file with the same name as the input or there is no input given at all, the program will read from stdin and return it to stdout.

Program Design

The main() function will be at the top of the program, with the three main function prototypes just above it so they can be referenced in main() with no issues. All program mechanisms will be found in main(). Under main(), I will fully declare and define the logic for each of the three functions I plan to utilize. The three main functions split this program into three sections, one part that takes the input, one part that checks the input, and the last part that decides what to do with the input depending on if it is valid or not.

Pseudocode

For this program, I plan to utilize several functions. These functions will take in the input, check if the input matches a readable file, and lastly print out the desired information from the file.

References

[1] is it positive or negative? how do you know?
[2] remember that negative numbers in 2s compliment convert things differently from positive numbers
[3] it is not the same as the `%x` format specifier