CSC 1600-001 Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

100 points Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Programming Assignment #5

Permissions

Due: Dec. 2, 2010

# Objective:

The purpose of this assignment is to provide you with the opportunity to familiarize yourself with the bit and shift operators. You will do this by reading and decoding a file containing zeros and ones.

# Notes:

You are to complete this assignment on your own. You will work on felix in your csc1600 directory. Your do not need to hand in anything. I will retrieve your completed code from your csc1600 directory for grading. Do not modify this code once the due date has passed. I will post any clarifications or revisions to the assignment on Web-CT.

**Assignment:**

Code a program, decode.c, in your CSC1600 directory. Your program will be reading and decoding files containing zeros and ones. Each file will contain an unspecified number of data lines. Each line will contain information on one file. The first nine characters on each line will specify the permissions for the file. The remainder of the characters will specify the file name. Your code should proceed in the following manner.

* Ask for an input file name.
* Read one line of data from the input file. (You can assume that a line of data will contain no more than 256 digits.)
* Decode the line of data and display the file name and permissions.
* Ask the user if there is a second file to decode. If yes, repeat the following steps.

**Data Assumptions:**

* Assume the first nine digits of each line represent three octal numbers between 0 and 7, the file permissions. (Example: The number 110100100 represents 644 and 111111111 represents 777.)
* Assume the remaining digits represent the ascii file name. (Keep in mind that a single ascii character is represented by a three-digit number between 0 and 127. Example: An ascii A is equivalent to a decimal 65, octal 101 or binary 001000001.)
* The following input produces the following output.

110110110001100110001101001001101100001100101

111111111001100110001101001001101100001100101000110001

111100100001100110001101001001101100001100101000110001000101110001100100001100001001110100

111100100001100110001101001001101100001100101000110010000101110001100100001100001001110100

What file would you like to process?

Binary.dat

file 666

file1 777

file1.dat 744

file2.dat 744

Would you like to read another file? (yes or no)

no

Where the third and fourth line are actually one line in the file as are the fourth and fifth.

* The following table shows a sample translation.

|  |  |
| --- | --- |
| 110 110 110 | 001 100 110 001 101 001 001 101 100 001 100 101 |
| 6 6 6 | 1 4 6 1 5 1 1 5 4 1 4 5 |
|  | f i l e |

* I have copied binary.dat to /mnt/a/csc1600. Use this file to test your code. Try making your own test file. I will use my own file to test your code.

**Hints:**

* You can assume that a file name will contain only alphanumeric characters, underscores and dots.
* You should be able to find a table of ascii characters on the internet.
* The digits 0 and 1 in your file are not binary numbers but ascii characters. 0 is stored as a

decimal 48 and 1 as a decimal 49.

* As I am asking you to display file permissions as an octal number, remember that it is easy to convert a binary number to an octal number using the shift operator
* Stick to bitwise and shift operators wherever you can.

# Evaluation

**I will calculate your grade as follows:**

* Correctness (70 points)

I will award you 10 points for correct execution of your main loop.

I will award you 20 points for each correct execution. I will give partial credit.

* Style (30 points)

Remember to identify yourself as the author of your code.

Comment your code.

Indent your code.

Eliminate all warnings.

Follow the directions.