# TCES 201
# Introduction to Computer Programming
# Homework 5 – Files, Strings, Arrays, Functions, etc.
# 20 Points

This homework tests your understanding of the topics covered so far in C Programming – Files, Strings, Arrays, Functions, etc.

In this assignment, you will write several functions as described below.

### *main*

This function will call the random_number_file generator function. If the call is successful, it will then call the hex_and_binary_file_generator function.

### *random_number_file_generator*

This function creates a new text file for writing called randomNumbers.txt. The function will use the rand() function to generate a number between 10 and 100. This will be the number of numbers to be generated and written to the file. Then in a for loop generate (randomly) that number of numbers, each number must be between 0 and 255 inclusive. Once a number is generated use the fprint() function to write it to the randomNumbers.txt file. Don't forget to append a newline ('\n') to the string so that every decimal number is on its own line. Once the loop completes close the file properly and return to the main function. This function must return 0 if successful, 1 if something went wrong in the code.

### *convert_decimal_to_hex*

This function takes a decimal number and converts it into hexadecimal string.

### *convert_decimal_to_binary*

This function takes a decimal number and converts it into binary string.

### hex_and_binary_file_generator

This function will open the randomNumbers.txt file for reading. It will also open a new file, hexAndBinary.txt for writing. The basic process is to read in one line of text from randomNumbers.txt. Use the fgets() function from stdio.h to read the string (http://www.acm.uiuc.edu/webmonkeys/book/c_guide/2.12.html#fgets). Then use the atoi() function from stdlib.h (http://www.acm.uiuc.edu/webmonkeys/book/c_guide/2.13.html#atoi) to convert that string to a decimal number. Once you have the decimal number, check to make sure it is between 0 and 255. If it is any other value you should print an error message to the screen and fprintf() a message to the file as well: "Error, number too large|small\n". If it is in the range then first convert it to hex and to binary. This will be fprinted to the file hexAndBinary.txt in the following format:

```
0x3A:  00111010  (58)
```

Call the convert_bin_to_hex and convert_dec_to_bin functions to convert the numbers to the different formats. You will have two strings, one hex and one binary (byte) so your format should look like this: "0x%s:  %s  (%d)\n" passing the arguments hex, byte, decimal number.

This will be done in a while loop that is looking for the <EOF> marker to terminate the loop. Use the feof() function for this purpose (http://www.acm.uiuc.edu/webmonkeys/book/c_guide/2.12.html#feof). When the loop terminates close the files properly and return to main.

The basic structure of your program should look like this:

```
/* Your header comments */
#includes
#defines

int main() {
    create and initialize needed variables - remember to consider if they need to be defined here
    or as locals in the other functions

    call your randomNumber file generator

    if no error call your hexAndBinary file generator

    clean up if necessary
}

int randomNumber file generator (parameters) {
    create and initialize variables
    open file
    for random number of times between 10 and 100
        do the work
        write to the file
    }
    clean up (close file)
}

int hexAndBinary file generator (parameters) {
    create and initialize variables
    open files
    while not the end of the input file
        read a number as a string and convert to decimal
        call necessary functions to convert to hex and binary strings
        write a composite string (as above) to the output file
    } when done
    close both files
}

void convert_decimal_to_hex (int decimalNo, char * hex) {
    convert decimalNo into hex.
}

void convert_decimal_to_binary (int decimalNo, char * byte) {
    convert decimalNo into a string of '1's and '0's (8)
}
```

**Submission Instructions:** Submit the code on Canvas under hw5 Submission link as convert_files.c. Formatting, appropriate variable names, readability and commenting are all considered while grading.