CMSC223 Systems Programming - Lab 3

I/O Redirection and Arrays Practice – Part 1

1. **SHOUTER:** Write a program in file `shout.c` to input whatever the user types on the keyboard and prints it out, all in uppercase letters. To understand the behavior of the program, try the command

   ```
   [xena@codewarriors cs223]$ cat
   ```

   When cat is used as above, without any file name, it simply outputs whatever you type in. It will not print out the characters as you type them, but only when you press the <ENTER> key. It will stop running once you type and <end-of-file> character. In the terminal, the <end-of-file> character is entered by typing <CTRL-D> on an empty line.

   Note that the program you are being asked is required to change all letters to uppercase. You can detect letters by using the `isalpha()` function, and you can use the `toupper()` function to convert alphabetic characters to uppercase. Consult your text for details on these functions.

   Extra: Instead of using the `toupper()` function, use arithmetic on characters to perform the conversion. See the section Character-Handling Functions on page 138 of King.

   You will use `getchar()` to read in individual characters and print them out. Your program should keep running until it reads <end-of-file>, which is indicated by a -1 (or EOF, as shown in class) return value from `getchar()`. Show the output of your program by first typing the following as input from the keyboard:

   ```
   She sells
   Sea shells
   By the seashore
   In Seychelles
   ```

   Next, create a text file `tse.txt` with the following contents:

   ```
   In the room women come and go
   Talking about Michaelangelo
   -: T. S. Eliot
   ```

   Run your program using I/O redirection as below:

   ```
   [xena@codewarriors cs223]$ ./shout < tse.txt
   ```

2. **HISTOGRAM:** Write a C program to print out a histogram from a set of input numbers. The input will be an undetermined number of scores on an exam (between 0..50, inclusive), one on each line, as shown below:

```
47
35
50
36
48
0
48
50
42
…
```

The output will be a histogram as shown below:

```
0: XXX
1: XXXX
2: X
…
48: XX
49:
50: XX
The average score for 43 students is 37.23.
```

Each X represents one occurrence of a score. Thus, in the above, there were three 0's in the input, two 48's, and no 49's, etc. The program also computes and prints out the average score, which is printed with exactly two digits after the decimal.

Hi**nts:** You will assume that input comes from standard input (keyboard) and output is to standard output. A data file is provided (use: `~dkumar/CMSC223/Lab3/exam7.txt`). You will include output from the data file in your submission. Use Linux I/O redirection to read the data from the file. Here is an algorithm for doing this:

```
Initialize an array of counts (from 0..50) to 0.
read the first score
while there is input to read
    record the input score in counts
    read next input
Output the histogram based on counts
```

You can add the steps needed to compute the average as needed.