Lab session (2nd Feb)

1. Easy

1.1. [Implementing library functions from math.h] Implement functions `my_floor`, `my_ceil` that work similar to `floor` and `ceil`.

1.2. [Limits using limits.h and float.h] Print the max and min values of various varieties of Integers and floats.

- Understand the meaning of these limits.
- Also understand the limits for (64-bit float) doubles.

1.3. [Implementing ctype.h functionalities] Read the functions that ctype.h header supports. Implement some of these functions on your own. Do it using functions.

2. Normal

2.1. [Implementing library functions from string.h] Implement functions `my_strlen`, `my_strcmp` and `my_strcpy`, that work similar to `strlen`, `strcmp` and `strcpy` respectively.

2.2. [Hackathon2 problem3 (substring search) using string.h] Solve the Hackathon#2 Problem#3 which checks for the patterns in the string using library functions available in string.h. Do it in two ways:

- [easier] using `strcmp`
- [slightly harder] using `strstr`.

2.3. Consider uppercase alphabet characters segregated into 3 sets:
[A-J] -> 1
[K-R] -> 2
[S-Z] -> 3
Implement a program that takes an arbitrary number of strings as command line arguments. For the nth string, the program prints the number of characters in the string that belong to the n%3 set.

Input: a.out AABQ SRDT
Output: 3 1
(AABQ -> 1st string, AAB belong to Set 1)
(SRDT -> 2nd string, R belongs to Set 2)

2.4. [Augment stdio.h to the input program] Read a inp.c program as input and add the #include `<stdio.h>\n` as the 1st line of the output-file.

2.5. Take a filename as (command-line) input. Then print the character count, word count, line count.

3. Learn by Experiments

3.1. Use the square-root functions provided by math.h and print the difference between the value computed by them and the values obtained by the two methods that we have discussed - bisection, Newton-Raphson.