Lab Assignment 2- Lexical Analysis (LEX/Flex)

```
Installing Flex: sudo apt-get update
                 sudo apt-get install flex
```

Editor for writing Lex/Flex program: Use any text editor

How to compile/execute: Check the lecture notes shared related to LEX/Flex tool.

SECTION 1

Q 1.1 Write a LEX/Flex program that recognizes binary strings containing at least six 1's.

Q 1.2 Write a Lex program that recognizes strings starting with a "c" and ending with an "r". Consider the alphabet to be the union of all the letters and digits.

Q 1.3 Write a LEX/Flex program that recognizes strings over the alphabet {a, b, c} where the length of the string should be 6, and the string should contain either two consecutive a's, or two consecutive b's.

SECTION 2

Q 2.1. Write a Lex program which identifies **Integers**, **float** numbers, **character constants** (any symbol within single quotes), and **String constants** (sequence of symbols within double quotes). Add the required rules to allow the lexical analyzer to ignore any illegal/unexpected symbol in the input and to continue the analysis. Your program should respond to various inputs as follows:

Sample Input	Sample lexical output
415	Integer
-12	Integer
899.16	Float
4#15	Integer (illegal symbol # will be ignored)
'b'	Character constant
"aa4$bb"	String constant

SECTION 3

Q 3. Construct a lexical analyzer for the following simple "**C**" like language using the Lex/Flex tool.

1. **Data Type** : integer (INT/int), floating point (FLOAT/float), character (CHAR/char), Boolean (Bool/bool)
2. **Condition constructs**: **if**
3. **Loop Constructs**: for, while
4. **Input / Output Constructs**:
    a. read(x) - Read into variable x
    b. out(x) - Write variable x to output

5. Relational operators, assignment and arithmetic operators
6. Only function is **main()**, there is no other function.

You may test it using the below example. Also, create other examples with while etc and test using them.

Example Input:

```
main()
{
        INT i=0;
        INT total=0;
        FLOAT count;
        CHAR ch;
        input(count);
        for(i=0;i<10;i++)
        {

               read(x);
               total= total+x;

        }

        out(total);
}
```
