**Language Processor (1st Lab Test in the week following 1st sessional)**

(Form groups of 2 within the same batch. These groups would be fixed for this semester.)

**Assignment 1: Lex and Yacc (NOTE: Lex can be replaced with Flex and Yacc can be replaced with Bison)**

**Write following Lex Programs (Input can be from “command line as string” or “file”)**

1. To count no of vowels and consonants.
2. To count the type of numbers-- +integer, -ve integer, +ve fraction, -ve fraction
3. To count the no of words, character, and lines.
4. To find if a character apart from alphabets occurs in a string.
5. To identify set of strings having 3 to 5 alphabets.

**Write following Yacc programs (Input can be from “command line as string” or “file”)**

1. Recognize all strings of the form anbn ; n>0
2. Recognize all strings that are palindromes
3. Accept all strings of balanced parenthesis
4. Create an evaluation tree and output the traversals for a given input string. The grammar to be used is

E -> E + E | E \* E | Int

Int -> 0|1|2|3|4|5|6|7|8|9

With \* having higher precedence than + and both operators are left-associative.

Also write programs to output pre-order/in-order/post-order traversals of this tree.

You should also be able to answer questions (like the following) that would be asked during viva:

1. What is a typical structure of a lex file and a yacc file?
2. How do you define global variables in lex and yacc?
3. What is ‘yytext’ and ‘yyleng’?
4. What is ‘yyless()’?
5. What regular expression you can use for finding compliment of a set?
6. What is ‘%token’?
7. How to associate a struct/union with non-terminals in YACC?
8. How to assign associativity and precedence to operators?
9. What is ‘yyerror()’?
10. How does lex and yacc work in sync?