CS F214
Logic in CS
BITS Pilani, Hyderabad Campus
Assignment -1
Due Date : 5th October 2016 (by Midnight)
Total Marks: 30 (weightage : 10%)

Objective: In this assignment, you have to implement the construction of a parse tree for a propositional logic formula. You can assume that the formula is well formed and fully parenthesized. The input to your program should be a propositional logical formula in infix notation. The connective symbols you will use for this assignment is as follows.

1. ~ for negation
2. + for OR
3. * for AND
4. > for implication.

Your code should have **separate functions** to perform the given tasks.

Task 1:
Write a function to **convert the infix** propositional logic expression **into a prefix** propositional logic expression. **[5]**

Task 2:
Write a function to **convert the prefix expression into a rooted binary parse tree**. **[5]**

Task 3:
Write a function to traverse the parse tree to **output the infix expression** back by in-order traversal of the parse tree. **[5]**

Task 4:
Write a function to **compute the height of a parse tree.** **[5]**

Task 5:
Write a function to **evaluate the truth value of a propositional logic formula**, given the truth values of each propositional atom by traversing the tree in a bottom up fashion. **[5]**

Task 6:
Analyze your algorithms in terms of output, efficiency, and memory usage. Make sure that your code is well indented and commented. Document your assignment as HTML pages (pay attention to the aesthetics). **[5]**

General Instructions:

1. This assignment will be done in groups of max three students.
2. **Code must be written in C or C++ only.**

3. You need to mail your working code and HTML pages in zip file to **rayt@hyderabad.bits-pilani.ac.in** by the deadline. Only one mail per group should be sent.
4. The name of the file should be **id1_LOGIC_A1.zip**, where id1 refers to the BITS ID of the sender.
5. **You can discuss with your friends but refrain from copying the code and submitting. Also please do not use code downloaded from internet. Such codes will receive 0 credits.**
6. You have to demo the code to the instructor on a scheduled date and timing after submission. **It is important to attend the demo, as absence from demo will amount to no credit for the assignment.**

References to look into:
http://scanftree.com/Data_Structure/infix-to-prefix

Algorithms in C++ by Robert Sedgewick