Class Exercise #4
COSC600 Advanced Data Structures and Algorithm Analysis

Name _____

1. Evaluate(Compute) the following postfix (reverse Polish) expression using a stack step-by-step).
   (Show the stack status when each operator is applied.)

   a) 4 2 + 3 * 4 6 2 / - 1 + 1 * /

   b) 3 6 7 4 - / + 5 2 1 3 2 - + * - +

2. Consider the following infix expressions:

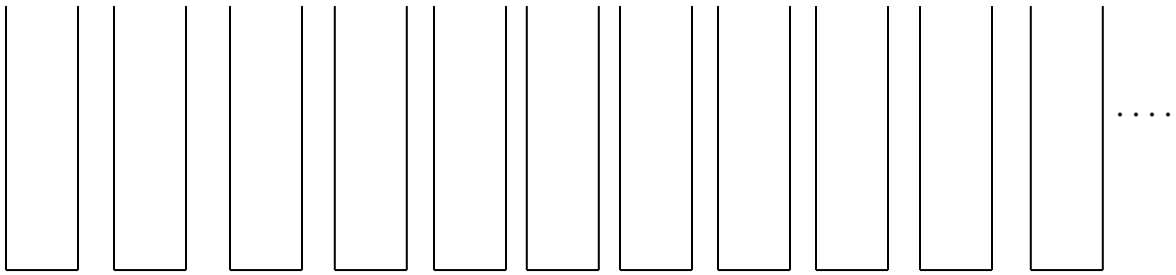
   1) 6 – 3 + 2 – 1 + 6 / 3 * 2 – 1 + (3 + 4 / 2 – 1)

   2) (5 - 12 / 3 + 2 * 6 / 3) + (4 * (8 - 5) / 2) + (6 + 10 / (8 – 3 * 2))

   3) (8 + 6 / (3 – 1) * 2 – 9 / (4 – 1)) – (4 – (8 – 2) / 3)

For each infix expression,

   a) show how to convert each infix expression to its postfix expression step-by-step. To show it, what is the stack status each time right after proceeding each operator (+, -, *, or /)? And what is the total number of push operations and pop operations done for each expression?



The postfix expression?

The total number of push operations? _____

The total number of pop operations? _____

b) based on your answers of a), trace the evaluation (computation) of the postfix expression using stack step-by step.

3. Write an algorithm in **pseudo-code** to convert a given positive integer decimal number, n, to a binary number using a stack, not using a recursive function.