Problem #1

In C++ implement a generic class, called Stack<T>, that uses a single-linked list implementation. It must implement the stack ADT. It must be generic on the type of the data to be stored. Give all class definitions and implement the following for Stack:

- Default constructor
- Destructor
- Copy constructor
- Swap – constant time swap (i.e., run time is not dependent on size of stack)
- Overload the assignment operator – using standard copy semantics
- push(T) – takes a parameter of type T and adds it to the stack
- T pop() – removes a node from the stack

You can NOT use STL or any other predefined library or built in types (such as std::string).

Problem #2

In C++ implement a binary search tree abstract data type (ADT) that uses dynamic memory allocation. Make it a tree of integers. Along with the class definition(s), you must implement the following methods for the class:

- Default constructor
- Destructor
- Copy constructor – must be recursive or use a recursive method to copy the nodes.
- insert which takes a parameter of type integer and creates a new node that is added to the tree in the correct position based on the rules of a binary search tree.

Your implementation can NOT use STL or any other libraries (standard or otherwise).
Problem #3

(a) Please convert each of the following two infix expressions into postfix and prefix notations.
A / B – C + D % E * F + H / I
A + (B + C) * (D – E) / F

(b) Give the preorder, postorder, and inorder traversals of the tree below: