Homework

Assignment 1 Write a C program allowing user to enter two polynomials. These polynomials are stored in single linked list.

a. Write a function to add two polynomials. Do not destroy the input. You must make sure that the output polynomial is sorted by exponent. If the polynomials have M and N terms, respectively, what is the time complexity of your program?
b. Write a function to multiply two polynomials. Notes: You must make sure that the output polynomial is sorted by exponent. In resulting polynomial, if the terms which have the same exponent should be combined. For example: x^2 + x + x + 1 => X^2 + 2X + 1.
If the polynomials have M and N terms, respectively, what is the time complexity of your program?

Assignment 2 Write functions to implement basic operations of a doubly linked list:

1. Create an empty doubly linked list

2. Insert a node into doubly linked list

3. Delete a node of doubly linked list

4. Find a node of doubly linked list

5. Sort the doubly linked list

After that writing a main function to test these functions. Assuming that doubly linked list will only contain integer.

Assignment 3 Write functions to implement basic operations of a stack:

1. Create an empty stack
2. isEmpty()
3. Pop an element at the top of a stack
4. Push an element at the top of a stack
5. Return value of the element at top of a stack

After that, using this stack:

a. write a C program to check for balancing symbols of an expression, supporting 3 kinds of symbols: (),[],{}.
b. Write a C program to evaluate a postfix expression
c. Write a C program to convert an infix expression which includes '(', ')', '+', '-', '*', '/' to postfix

Assignment 4 Write functions to implement basic operations of a queue:

1. Create an empty queue
2. isEmpy()
3. Insert an element at the tail of a queue
4. Delete an element at the head of a queue

After that, write a C program to test these functions. Assuming that the queue will only contain integer

Assignment 5 A deque is a data structure consisting of a list of items, on which the following operations are possible:

Push(X,D): Insert item X on the front end of deque D.

Pop(D): Remove the front item from deque D and return it.

Inject(X,D): Insert item X on the rear end of deque D.

Eject(D): Remove the rear item from deque D and return it.

Write a C program to support the deque that take 0(1) time per operation.