CST370 Midterm: Review Questions

Linked Lists

1. Write a Count() function that counts the number of times a given integer occurs in a list.
   e.g., the function Count() might have the following form:
   int Count(int searchFor)

2. Write a function GetNth( ) that takes a linked list and an integer index and returns the data value stored in the node at that index position. GetNth() uses the C numbering convention that the first node is index 0, the second is index 1, ... and so on. So for the list {42, 13, 666}, GetNth() with index 1 should return 13. The index should be in the range [0..length-1]. If it is not, GetNth() should print an error.
   e.g., the function GetNth() might look like the following form
   int GetNth(int index)

3. Write a Pop() function that takes a list, deletes the head node, and returns the head node's data. If the list is empty, it prints out "LIST IS EMPTY".

4. Write an Append() function that takes two lists, 'a' and 'b', appends 'b' onto the end of 'a'.

5. Write a function Duplicates( ) that takes a list containing integers, and returns the list after removing all duplicate entries from the list. Thus given the list 20, 25, 25, 30, 40, 45, 45, 45, 60, it should return 20, 25, 30, 40, 45, 60. Write a driver function to test all the above functions. You could use some of examples given here to test whether your functions are working correctly.

Stacks

In this question, you will determine whether an expression is balanced or not. For example, expressions ({[ ]}) and ({})[ ] are balanced, while expressions (}) and ({}[ ) is not. For determining if an expression is balanced or not you will need to use a stack. You can assume that your expression is a string and contains only (,), {, }, [, and]. Your program should return true if the expression is balanced and false if it is not. You can use the stack class provided to you in class for this question.

Queues

Write a driver function which starts with an empty queue and

a) enqueues 50 numbers 10, 20, 30 … 500 in order. Your program should then dequeue 10 items from the queue. Output the value element in the front of the queue.
b) enqueues 5 numbers [1, 3, -5, 6, -10] in order. Your program then dequeues 3 elements from the queue. Print out contents of the current queue. You can use the queue class provided to you in class for this question.