

Question 1:

(20 minutes) [30 pts]

When analyzing arithmetic expressions, it is important to determine whether an expression is balanced with respect to different kinds of parentheses. For example:

$$(a * [x + y] / z - [p / \{q - r\}])$$

is balanced, but the expression

$$(a * [x + y] / z - [p / \{q - r\}])$$

is not, because the sub-expression  $[x + y)$  is incorrect.

Write a C++ program that asks the user for an expression string and displays a message indicating whether the expression has balanced parentheses. The program should call a Boolean method, `isBalanced`, that takes as an input the expression string and returns true if the expression is balanced or false if it is not. Use stacks in your implementation.

Hint: `{`, `}`, `(`, `)`, `[`, `]` are the only symbols considered for the check. All other characters can be ignored.

**Question 2:**

**(25 minutes) [35 pts]**

Write a modular C++ program that reads a string of characters. The program should separate the characters found on the even positions in the string from those on the odd positions into two different linked lists. Next the program should let the user select between the following choices:

1. Process the lists.
2. Merge the lists in the increasing order.

If choice (1) is selected the user gets the following choices:

1. Print the number of characters in each list.
2. Check if the characters in the two lists are equal.
3. Check if the characters in the two lists form a palindrome.

If choice (2) is selected, merge the two linked lists into a new linked list.

Question 3:

(10 minutes) [10 pts]

- a- How to implement a 2-dimensional array in main memory? Suggest a **general formula** to get the location of element (i,j) in a column-Based representation given the location of the first entry in this array.
  
- b- What are the advantages of a linked list over an array implementation?

Question 4:

(20 minutes) [25 pts]

Write the needed C++ code that searches a double linked list with first node pointed to by pointer '**first**' for a given item and, if the item is found, return a pointer to the node containing that item.