Lab # 5

Task 1:

Write an Assembly Language Program that lets the user to Enter 5 binary numbers 8 bit each, store these numbers in an array and then display these numbers in Hexadecimal form on console. Design following two procedures for this task:

1. **"BinaryInput"** to get a single 8-bit number from user.
2. **"HexadecimalOutput"** to display a single 8-bit hexadecimal number on console.

Use Stack for Task 2 to Task 5

Task 2:

Write an Assembly Language Program that lets the user type some text, consisting of words separated by blanks, ending with a carriage return, and displays the text in the same word ordered as entered, but with the letters in each word reversed.

Sample execution:

Enter a string: **COAL is an interesting Subject**

You Enter: **LAOC si na gnitseretni tcejbuS**

Task 3:

Write an Assembly Language Program that lets the user type in an algebraic expression, ending with a carriage return that contains parenthesis only. As the expression is being typed in, the program evaluates each character. If at any point the expression is incorrectly bracketed [too many left or right], the program tells the user to start over. After the carriage return is typed, if the expression is correct, the program displays "expression is correct" and the program asks the user if he or she wants to continue. If the user types 'Y', the program runs again. If the expression is not correct, the program displays "too many left brackets, begin again" or "too many right brackets, begin again" according to expression. Your program does not need to store the input string, only check it for correctness.

Sample execution:

Enter an algebraic expression: **a+b)**

Too many right brackets, Begin again

Enter an algebraic expression: **a+(b-c**

Too many left brackets, Begin again

Enter an algebraic expression: **a+(b-c)**

"Expression is correct"

Type Y if you want to continue: **N**

Task 4:

Write a program to check whether a given string is palindrome or not.

Sample execution:

Input String: "abba"

Output: String is palindrome

Input String: "abbca"

Output: String is not palindrome

Task 5:

You are given a string consisting of lowercase English letters. A duplicate removal consists of choosing two adjacent and equal letters and removing them.

We repeatedly make duplicate removals on string until we no longer can.

Return the final string after all such duplicate removals have been made.

Sample execution:

Example 1:

Input: s = "abbaca"

Output: "ca"

Explanation:

For example, in "abbaca" we could remove "bb" since the letters are adjacent and equal, and this is the only possible move. The result of this move is that the string is "aaca", of which only "aa" is possible, so the final string is "ca".

Example 2:

Input: s = "azxxzy"

Output: "ay"