1. **Stack Class (STACK):**
   * Represents a stack data structure to store characters.
   * Private members include an array **a** to store characters and an integer **top** to keep track of the top of the stack.
2. **Methods of STACK Class:**
   * **push(char c) Method:**
     + Pushes a character onto the stack.
     + Increments the **top** and assigns the character to the corresponding position in the array.
   * **reverse() Method:**
     + Prints the reversed string by popping characters from the stack.
     + Copies characters to a new array (**str**) to preserve the original order.
   * **convert(char str[]) Method:**
     + Converts the input string to lowercase, removing non-alphabetic characters.
     + Updates the original string (**str**) with the converted string.
     + Displays the converted string.
   * **palindrome() Method:**
     + Checks whether the string is a palindrome or not.
     + Compares the reversed string with the original string and prints the result.
3. **main Function:**
   * Creates an instance of the **STACK** class named **stack**.
   * Takes a string as input from the user.
   * Calls the **convert** method to convert the string to lowercase and remove non-alphabetic characters.
   * Pushes each character of the string onto the stack.
   * Calls the **palindrome** method to check if the string is a palindrome.
   * Calls the **reverse** method to display the reversed string.
4. **User Interaction:**
   * The program interacts with the user by taking a string as input.
   * It performs operations like converting to lowercase, checking for palindrome, and displaying the reversed string.
5. **Note:**
   * The code demonstrates the usage of a stack to reverse a string, convert it to lowercase, and check if it is a palindrome. The stack helps in reversing the string by using the Last In First Out (LIFO) property of the stack. The program uses a simple stack-based approach for string manipulation and checking.

Algorithm:

1. \*Include necessary libraries and define the maximum size of the stack.\*

2. \*\*Create a class named STACK to encapsulate stack-related operations.\*\*

3. \*\*Implement the member functions of the STACK class: push, reverse, convert, and palindrome.\*\*

4. \*\*In the main function, create an instance of the STACK class, input a string, and perform the stack operations.\*\*

Top of Form