**Institute of Computer Technology**

**B. Tech. Computer Science and Engineering**

**Sub: DS**

**Course Code: 2CSE302**

**Practical – 5**

**Name: Jaymin Gondaliya**

**Enrollment No: 23162171007**

**Sem - 3**

**Branch: CS**

**Class: A**

**Batch: 32**

**Problem Definition-1:** Implement an Undo Mechanism in a Text Editor

**Code:**

#include <stdio.h>

#include <string.h>

#define MAX\_ACTIONS 10

#define MAX\_TEXT\_LENGTH 100

*// Structure to hold an action and the corresponding text*

typedef struct {

    char text[MAX\_TEXT\_LENGTH];

} Action;

Action undoStack[MAX\_ACTIONS]; *// Array to store undo actions*

int stackTop = -1; *// Index of the top element in the stack*

char currentText[MAX\_TEXT\_LENGTH] = ""; *// Current text in the editor*

*// Function to check if the undo stack is empty*

int isStackEmpty() {

    return stackTop == -1;

}

*// Function to save the current text to the stack*

void saveState() {

    if (stackTop < MAX\_ACTIONS - 1) {

        stackTop++;

        strcpy(undoStack[stackTop].text, currentText);

    } else {

        printf("Undo stack is full!\n");

    }

}

*// Function to undo the last action*

void undo() {

    if (!isStackEmpty()) {

        stackTop--;

        if (stackTop >= 0) {

            strcpy(currentText, undoStack[stackTop].text);

        } else {

            currentText[0] = '\0'; *// Clear current text if no undo available*

        }

    } else {

        printf("Undo stack is empty!\n");

    }

    printf("Text after undo: %s\n", currentText);

}

*// Function to show the last text saved without removing it*

void peekNextUndo() {

    if (!isStackEmpty()) {

        printf("Next undo text: %s\n", undoStack[stackTop].text);

    } else {

        printf("Undo stack is empty!\n");

    }

}

*// Function to show all texts saved in the stack*

void showAllTexts() {

    if (stackTop == -1) {

        printf("No texts typed yet.\n");

    } else {

        printf("All saved texts:\n");

        for (int i = 0; i <= stackTop; i++) {

            printf("%s\n", undoStack[i].text);

        }

    }

}

*// Main function to provide a menu for text editing and undo operations*

int main() {

    int choice;

    do {

        printf("\nMenu:\n");

        printf("1. Type Text\n");

        printf("2. Undo Last\n");

        printf("3. Peek Next Undo\n");

        printf("4. Show All Texts\n");

        printf("5. Exit\n");

        printf("Enter your choice: ");

        scanf("%d", &choice);

        char inputText[MAX\_TEXT\_LENGTH];

        switch (choice) {

            case 1: *// Typing new text*

                printf("Enter text: ");

                scanf(" %[^\n]s", inputText);

                strcpy(currentText, inputText); *// Save the new text as current text*

                saveState(); *// Save the state after typing*

                printf("Current Text: %s\n", currentText);

                break;

            case 2: *// Undo last action*

                undo();

                break;

            case 3: *// Peek at the next undo action*

                peekNextUndo();

                break;

            case 4: *// Show all texts saved*

                showAllTexts();

                break;

            case 5: *// Exit the program*

                printf("Exiting...\n");

                break;

            default: *// Invalid menu choice*

                printf("Invalid choice!\n");

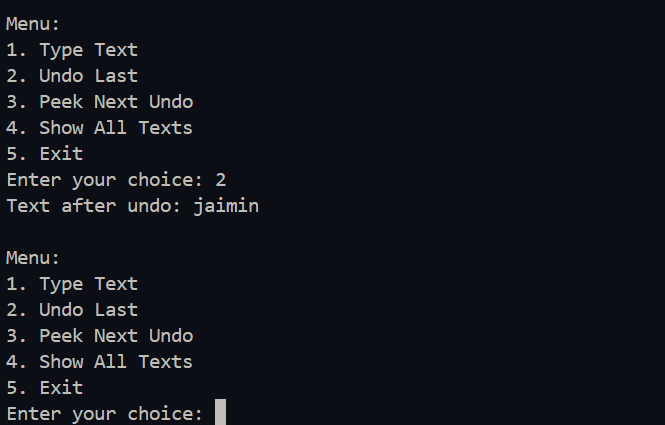
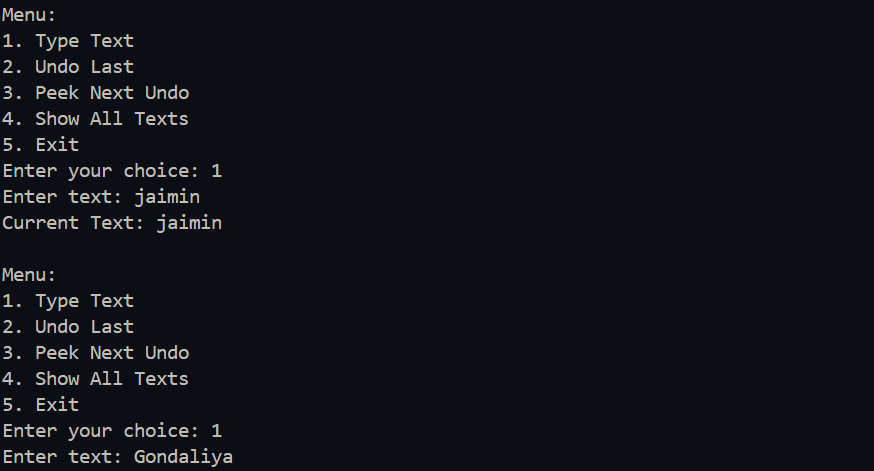
        }

    } while (choice != 5);

    return 0;

}

**Output-**

****