Practical No 7

- Program 1 : Implementation of STACK. To perform any Stack operations.

```
//Name : Rakesh Mahadev Bandi
//Roll No : 3
//Class : SYCSE
//PRN No : 2024065738

#include <stdio.h>
#include <stdlib.h>

#define MAX 5

int stack[MAX];
int top = -1;

void push(int value) {
   if (top == MAX - 1) {
      printf("Stack Overflow!\n");
   } else {
      stack[++top] = value;
      printf("%d pushed onto the stack.\n", value);
   }
}

int pop() {
   if (top == -1) {
      printf("Stack Underflow!\n");
      return -1;
   } else {
      return stack[top--];
   }
}

void display() {
   if (top == -1) {
      printf("Stack is empty.\n");
   } else {
      printf("Stack elements: ");
```

```
        int i;
        for (i = top; i >= 0; i--) {
            printf("%d ", stack[i]);
        }
        printf("\n");
    }
}


int main() {
    int choice, value;

    while (1) {
        printf("\nStack Operations Menu:\n");
        printf("1. Push\n");
        printf("2. Pop\n");
        printf("3. Display\n");
        printf("4. Exit\n");
        printf("Enter your choice (1-4): ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter a value to push: ");
                scanf("%d", &value);
                push(value);
                break;
            case 2:
                value = pop();
                if (value != -1) {
                    printf("Popped %d from the stack.\n", value);
                }
                break;
            case 3:
                display();
                break;
            case 4:
                printf("Exiting program.\n");
                exit(0);
            default:
```

```
            printf("Invalid choice. Please select a valid option (1-4).\n");
        }
    }

    return 0;
}
```

Output: