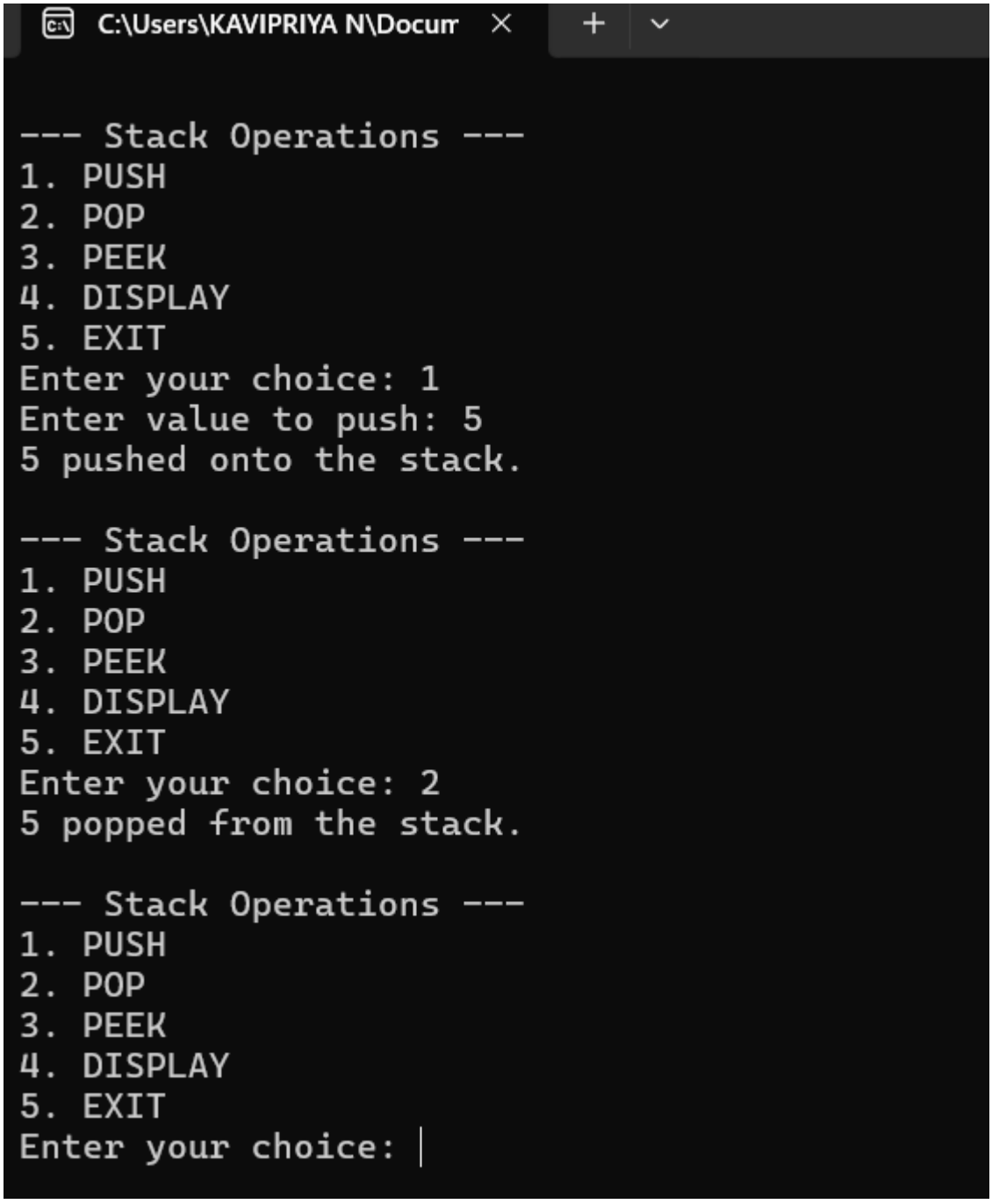
```
#include <stdio.h>
#include <stdlib.h>
#define MAX 100

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

// Function to push an element onto the stack
void push(int value) {
    if (top == MAX - 1) {
        printf("Stack Overflow! Cannot push %d\n", value);
    } else {
        stack[++top] = value;
        printf("%d pushed onto the stack.\n", value);
    }
}

// Function to pop an element from the stack
void pop() {
    if (top == -1) {
        printf("Stack Underflow! Cannot pop.\n");
    } else {
        printf("%d popped from the stack.\n", stack[top--]);
    }
}

// Function to peek at the top element of the stack
void peek() {
    if (top == -1) {
        printf("Stack is empty.\n");
    } else {
```

```
        printf("Top element is %d\n", stack[top]);
    }
}

// Function to display all elements of the stack
void display() {
    if (top == -1) {
        printf("Stack is empty.\n");
    } else {
        printf("Stack elements (top to bottom):\n");
        for (int i = top; i >= 0; i--) {
            printf("%d\n", stack[i]);
        }
    }
}

// Main menu-driven program
int main() {
    int choice, value;

    while (1) {
        printf("\n--- Stack Operations ---\n");
        printf("1. PUSH\n");
        printf("2. POP\n");
        printf("3. PEEK\n");
        printf("4. DISPLAY\n");
        printf("5. EXIT\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
```

```
            case 1:
                printf("Enter value to push: ");
                scanf("%d", &value);
                push(value);
                break;
            case 2:
                pop();
                break;
            case 3:
                peek();
                break;
            case 4:
                display();
                break;
            case 5:
                printf("Exiting program.\n");
                exit(0);
            default:
                printf("Invalid choice! Try again.\n");
        }
    }

    return 0;
}
```


C:\Users\KAVIPRIYA N\Docum
--- Stack Operations ---
1. PUSH
2. POP
3. PEEK
4. DISPLAY
5. EXIT
Enter your choice: 1
Enter value to push: 5
5 pushed onto the stack.
--- Stack Operations ---
1. PUSH
2. POP
3. PEEK
4. DISPLAY
5. EXIT
Enter your choice: 2
5 popped from the stack.
--- Stack Operations ---
1. PUSH
2. POP
3. PEEK
4. DISPLAY
5. EXIT
Enter your choice: