

Name-Jeevan R

Sec- I

DSA in C

Lab Program 1: Write a program to simulate the working of stack using an array with the following:

a) Push

b) Pop

c) Display

The program should print appropriate messages for stack overflow, stack underflow

Code:

```
#include <stdio.h>
```

```
#define SIZE 5
```

```
int stack[SIZE];
```

```
int top = -1;
```

```
// Function to push an element into the stack
```

```
void push(int value) {
```

```
    if (top == SIZE - 1) {
```

```
        printf("Stack Overflow! Cannot push %d\n", value);
```

```
    } else {
```

```
        top++;
```

```
        stack[top] = value;
```

```
        printf("%d pushed into stack\n", value);
```

```
    }
```

```
}
```

```
// Function to pop an element from the stack
```

```
void pop() {
```

```

if (top == -1) {
    printf("Stack Underflow! No element to pop\n");
} else {
    printf("%d popped from stack\n", stack[top]);
    top--;
}
}

```

// Enhanced function to display stack elements

```

void display() {
    if (top == -1) {
        printf("Stack is empty!\n");
    } else {
        printf("\nCurrent Stack (Top to Bottom):\n");
        for (int i = top; i >= 0; i--) {
            if (i == top) {
                printf("| %d | <- Top\n", stack[i]);
            } else {
                printf("| %d |\n", stack[i]);
            }
        }
        printf(" ---- \n");
    }
}

```

// Main function to run the menu-driven stack program

```

int main() {
    int choice, value;

    while (1) {
        printf("\n---- Stack Menu ----\n");

```

```

printf("1. Push\n");
printf("2. Pop\n");
printf("3. Display Stack\n");
printf("4. 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:
        display();
        break;
    case 4:
        printf("Exiting program.\n");
        return 0;
    default:
        printf("Invalid choice! Try again.\n");
}
}

return 0;
}

```

Expected Output:

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 3

Stack is empty!

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 1

Enter value to push: 10

10 pushed into stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 1

Enter value to push: 20

20 pushed into stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 1

Enter value to push: 30

30 pushed into stack

---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 1

Enter value to push: 40

Stack Overflow! Cannot push 40

---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 3

Current Stack (Top to Bottom):

| 30 | <- Top

| 20 |

| 10 |

---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 2

30 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

20 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

10 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

Stack Underflow! No element to pop

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 4

Exiting program.

=== Code Execution Successful ===---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 3

Stack is empty!

---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 1

Enter value to push: 10

10 pushed into stack

---- Stack Menu ----

1. Push

2. Pop

3. Display Stack

4. Exit

Enter your choice: 1

Enter value to push: 20

20 pushed into stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 1

Enter value to push: 30

30 pushed into stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 1

Enter value to push: 40

Stack Overflow! Cannot push 40

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 3

Current Stack (Top to Bottom):

| 30 | <- Top

| 20 |

| 10 |

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

30 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

20 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

10 popped from stack

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 2

Stack Underflow! No element to pop

---- Stack Menu ----

1. Push
2. Pop
3. Display Stack
4. Exit

Enter your choice: 4

Exiting program.

=== Code Execution Successful ===

Execution Images:

```
main.c
33 printf("\nCurrent Stack (Top to Bottom):\n");
34 for (int i = top; i >= 0; i--) {
35     if (i == top) {
36         printf("| %d | <- Top\n", stack[i]);
37     } else {
38         printf("| %d |\n", stack[i]);
39     }
40 }
41 printf("----- \n");
42 }
43 }
44
45 // Main function to run the menu-driven stack program
46 int main() {
47     int choice, value;
48
49     while (1) {
50         printf("\n---- Stack Menu ----\n");
51         printf("1. Push\n");
52         printf("2. Pop\n");
53         printf("3. Display Stack\n");
54         printf("4. Exit\n");
55         printf("Enter your choice: ");
56         scanf("%d", &choice);
57
58         switch (choice) {
59             case 1:
60                 printf("Enter value to push: ");
61                 scanf("%d", &value);
62                 push(value);
63                 break;
64             case 2:
65                 pop();
66         }
67     }
68 }
```

Output

Enter value to push: 30
30 pushed into stack

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 1
Enter value to push: 40
Stack Overflow! Cannot push 40

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 3

Current Stack (Top to Bottom):
| 30 | <- Top
| 20 |
10

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 2
30 popped from stack

```
main.c
1 #include <stdio.h>
2 #define SIZE 3
3
4 int stack[SIZE];
5
6 int top = -1;
7 // Function to push an element into the stack
8 void push(int value) {
9     if (top == SIZE - 1) {
10         printf("Stack Overflow! Cannot push %d\n", value);
11     } else {
12         top++;
13         stack[top] = value;
14         printf("%d pushed into stack\n", value);
15     }
16 }
17
18 // Function to pop an element from the stack
19 void pop() {
20     if (top == -1) {
21         printf("Stack Underflow! No element to pop\n");
22     } else {
23         printf("%d popped from stack\n", stack[top]);
24         top--;
25     }
26 }
27
28 // Enhanced function to display stack elements
29 void display() {
30     if (top == -1) {
31         printf("Stack is empty!\n");
32     } else {
33         printf("\nCurrent Stack (Top to Bottom):\n");
34         for (int i = top; i >= 0; i--) {
35             if (i == top) {
36                 printf("| %d | <- Top\n", stack[i]);
37             } else {
38                 printf("| %d |\n", stack[i]);
39             }
40         }
41         printf("----- \n");
42     }
43 }
```

Output

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 3
Stack is empty!

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 1
Enter value to push: 10
10 pushed into stack

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 1
Enter value to push: 20
20 pushed into stack

---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 1

main.c

Run

Clear

```
48
49 while (1) {
50     printf("\n---- Stack Menu ----\n");
51     printf("1. Push\n");
52     printf("2. Pop\n");
53     printf("3. Display Stack\n");
54     printf("4. Exit\n");
55     printf("Enter your choice: ");
56     scanf("%d", &choice);
57
58     switch (choice) {
59         case 1:
60             printf("Enter value to push: ");
61             scanf("%d", &value);
62             push(value);
63             break;
64         case 2:
65             pop();
66             break;
67         case 3:
68             display();
69             break;
70         case 4:
71             printf("Exiting program.\n");
72             return 0;
73         default:
74             printf("Invalid choice! Try again.\n");
75     }
76 }
77
78 return 0;
79 }
80
```

Output

```
Enter your choice: 1
30 popped from stack
---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 2
20 popped from stack
---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 2
10 popped from stack
---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 2
Stack Underflow! No element to pop
---- Stack Menu ----
1. Push
2. Pop
3. Display Stack
4. Exit
Enter your choice: 4
Exiting program.
```