**WEEK 01   
Stack implementation using arrays :**

**Code:**

#include <stdio.h>

#include <stdlib.h>

#define SIZE 3

int top = -1;

int stack[SIZE];

void push(int item) {

if (top == SIZE - 1) {

printf("\nStack overflow");

} else {

top++;

stack[top] = item;

printf("\nElement %d pushed to stack", item);

}

}

void pop() {

if (top == -1) {

printf("\nStack underflow");

} else {

printf("\nElement popped is %d", stack[top]);

top--;

}

}

void display() {

if (top == -1) {

printf("\nStack is empty");

} else {

printf("\nStack values:");

for (int i = top; i >= 0; i--) {

printf("\n%d", stack[i]);

}

}

}

int main() {

int ch, item;

for (;;) {

printf("\n\n1: Push");

printf("\n2: Pop");

printf("\n3: Display");

printf("\n4: Exit");

printf("\nEnter your choice: ");

scanf("%d", &ch);

switch (ch) {

case 1:

printf("Enter value to be pushed: ");

scanf("%d", &item);

push(item);

break;

case 2:

pop();

break;

case 3:

display();

break;

case 4:

exit(0);

break;

default:

printf("\nInvalid choice. Please try again.");

break;

}

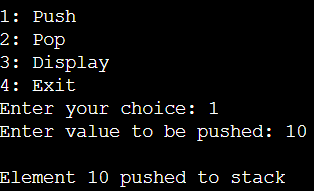
}

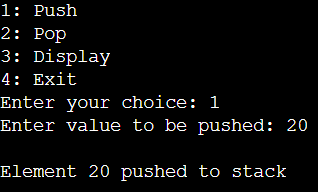
return 0;

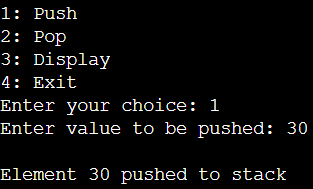
}

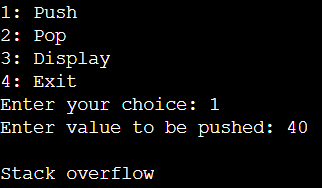
**Output :**

**Pushing values:**

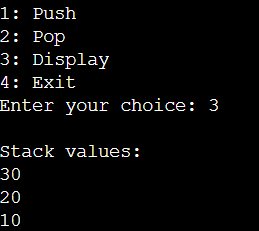


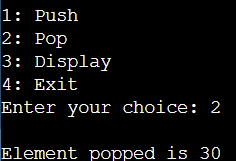


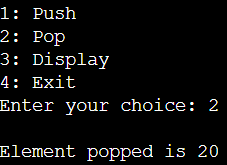


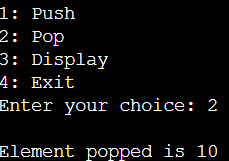


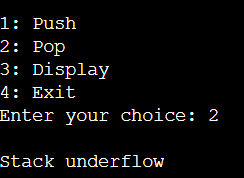
**Poping values:**











**Display:**

