```c
#include <stdio.h>
#define SIZE 20
int array[SIZE];        // declaration of array type variable.
int top1 = -1;
int top2 = SIZE;

//Function to push data into stack1
void push1 (int data)
{
// checking the overflow condition
  if (top1 < top2 - 1)
    {
      top1++;
      array[top1] = data;
    }
  else
    {
      printf ("Stack is full");
    }
}

// Function to push data into stack2
void push2 (int data)
{
// checking overflow condition
  if (top1 < top2 - 1)
    {
      top2--;
      array[top2] = data;
    }
  else
    {
      printf ("Stack is full..\n");
    }
}

//Function to pop data from the Stack1
void pop1 ()
{
```

```c
// Checking the underflow condition
  if (top1 >= 0)
    {
      int popped_element = array[top1];
      top1--;

      printf ("%d is being popped from Stack 1\n", popped_element
    }
  else
    {
      printf ("Stack is Empty \n");
    }
}

// Function to remove the element from the Stack2
void pop2 ()
{
// Checking underflow condition
  if (top2 < SIZE)
    {
      int popped_element = array[top2];
      top2--;

      printf ("%d is being popped from Stack 1\n", popped_element
    }
  else
    {
      printf ("Stack is Empty!\n");
    }
}

//Functions to Print the values of Stack1
void display_stack1 ()
{
  int i;
  for (i = top1; i >= 0; --i)
    {
      printf ("%d ", array[i]);
    }
  printf ("\n");
}

// Function to print the values of Stack2
void display_stack2 ()
{
  int i;
  for (i = top2; i < SIZE; ++i)
    {
      printf ("%d ", array[i]);
    }
  printf ("\n");
}

int main ()
{
  int ar[SIZE];
  int i;
  int num_of_ele;

  printf ("We can push a total of 20 values\n");

  //Number of elements pushed in stack 1 is 10
  //Number of elements pushed in stack 2 is 10

// loop to insert the elements into Stack1
  for (i = 1; i <= 10; ++i)
    {
      push1 (i);
      printf ("Value Pushed in Stack 1 is %d\n", i);
    }
// loop to insert the elements into Stack2.
  for (i = 11; i <= 20; ++i)
    {
      push2 (i);
      printf ("Value Pushed in Stack 2 is %d\n", i);
    }

  //Print Both Stacks
  display_stack1 ();
  display_stack2 ();

  //Pushing on Stack Full
  printf ("Pushing Value in Stack 1 is %d\n", 11);
  push1 (11);

  //Popping All Elements from Stack 1
  num_of_ele = top1 + 1;
  while (num_of_ele)
    {
      pop1 ();
      --num_of_ele;
    }

  // Trying to Pop the element From the Empty Stack
  pop1 ();

  return 0;
}
```