**Experiment 3**

**Aim:**Implement two stacks using a single array.

**Algorithm:**

1.Declare the size of the array arr.

2. Divide this array in two equal parts consisting of the same no. of elements represented as stacks.

3. Implement the functions for push and pop for both the stacks checking the overflow condition for push and underflow condition for pop.

4. Push elements in both the stacks.

5. Display both the stacks stack1 and stack2.

6. Pop elements in a stack1 to make it empty.

**Source Code:**

#include <stdio.h>

int main(){

    int n, a[100] = {0}, top1, top2, st1, st2, en1, en2, o, sta;

    printf("Size of the array: ");

    scanf("%d", &n);

    top1 = -1, top2 = n;

    st1 = 0, en1 = n / 2, en2 = en1 + 1, st2 = n - 1;

    for (int x = 0; x < 100; x++)   {

        printf("Press 1 for stack 1 or 2 for stack 2 or 3 to exit: ");

        scanf("%d", &sta);

        if (sta == 1) {

            printf("Stack 1: \n");

            for (int j = 0; j < 100; j++)    {

                printf("Press 1 to push, 2 to pop, 3 to exit stack: ");

                scanf("%d", &o);

                if (o == 1)     {

                    if (top1 >= en1 - 1)

                    {

                        printf("Overflow in Stack 1 \n");

                    }

                    else

                    {

                        top1++;

                        printf("Element to be pushed: ");

                        scanf("%d", &a[top1]);

                        printf("Stack 1: \n");

                        for (int i = st1; i < en1; i++)

                        {

                            printf("%d\t", a[i]);

                        }

                        printf("\n");

                    }

                }

                else if (o == 2)

                {

                    if (top1 == -1)

                    {

                        printf("Underflow in Stack 1 \n");

                    }

                    else

                    {

                        printf("%d popped!\n", a[top1]);

                        a[top1] = 0;

                        top1 -= 1;

                        for (int i = st1; i < en1; i++)

                        {

                            printf("%d\t", a[i]);

                        }

                        printf("\n");

                    }

                }

                else if (o == 3)

                {

                    printf("Exiting stack 1!\n");

                    break;

                }

                else

                {

                    printf("Error!\n");

                }

            }

        }

        else if (sta == 2)

        {

            printf("Stack 2: \n");

            for (int j = 0; j < 100; j++)

            {

                printf("Press 1 to push, 2 to pop, 3 to exit stack: ");

                scanf("%d", &o);

                if (o == 1)

                {

                    if (top2 == en1)

                    {

                        printf("Overflow in Stack 2 \n");

                    }

                    else

                    {

                        top2--;

                        printf("Element to be pushed:");

                        scanf("%d", &a[top2]);

                        printf("Stack 2: \n");

                        for (int y = st2; y >= en2-1; y--)

                        {

                            printf("%d\t", a[y]);

                        }

                        printf("\n");

                    }

                }

                else if (o == 2){

                    if (top2 == n)

                    {

                        printf("Underflow in Stack 2 \n");

                    }

                    else

                    {

                        printf("%d popped!\n", a[top2]);

                        a[top2] = 0;

                        top2++;

                        for (int y = st2; y >= en2-1; y--)

                        {

                            printf("%d\t", a[y]);

                        }

                        printf("\n");

                    }

                }

                else if (o == 3)

                {

                    printf("Exiting stack 2!\n");

                    break;

                }

                else

                {

                    printf("Error!\n");

                }

            }

        }

        else if (sta == 3)

        {

            printf("Exiting array!\n");

            break;

        }

    }

}

**Output:**

