**Assignment: 8**

**Exp.1.** Write code to implement Stack Data structures. Implement the Push and Pop operation in the stack.

**Coding:**

#include<stdio.h>

#include<stdlib.h>

#define size 5

int top=-1,ar[size];

void Push();

void Pop();

void show();

int main()

{

    int choice;

    while(1)

    {

        printf("\nOperations performed by Stack");

        printf("\n1.Push the element\n2.Pop the element\n3.Show\n4.End");

        printf("\n\nEnter your choice: ");

        scanf("%d",&choice);

        switch(choice)

        {

            case 1: Push();

                    break;

            case 2: Pop();

                    break;

            case 3: show();

                    break;

            case 4: exit(0);

            default: printf("\nInvalid choice!!");

        }

    }

    return 0;

}

void Push()

{

    int num;

    if(top==size-1)

    {

        printf("\nStack Overflowed");

    }

    else

    {

        printf("\nEnter Number To Be Added To Stack: ");

        scanf("%d",&num);

        top=top+1;

        ar[top]=num;

    }

}

void Pop()

{

    if(top==-1)

    {

        printf("\nStack Is Now Empty");

    }

    else

    {

        printf("\nElement Popped is %d",ar[top]);

        top=top-1;

    }

}

void show()

{

    int i;

    if(top==-1)

    {

        printf("\nThe Stack Is empty");

    }

    else

    {

        printf("\nElements Of The Stack are: \n");

        for(i=top;i>=0;i--)

        {

            printf("%d",ar[i]);

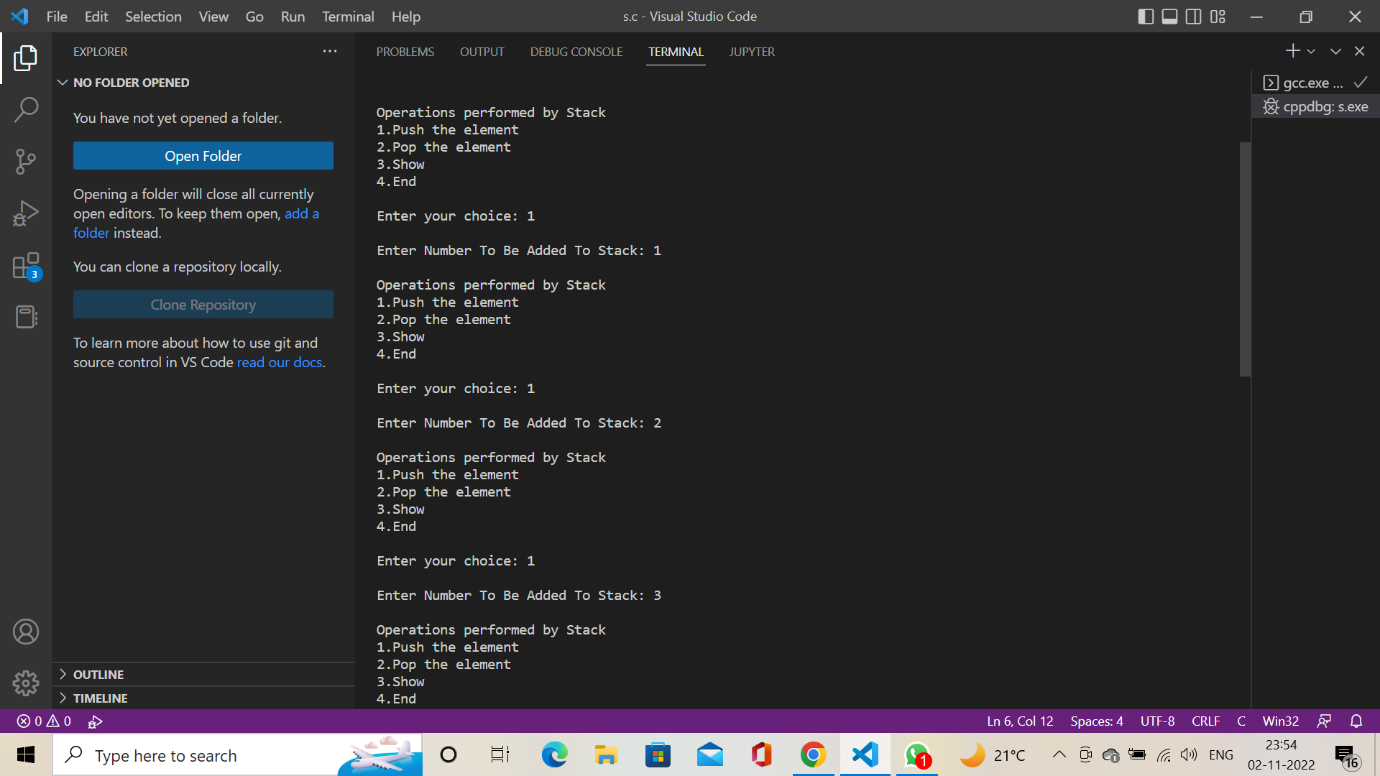
            printf("\n");

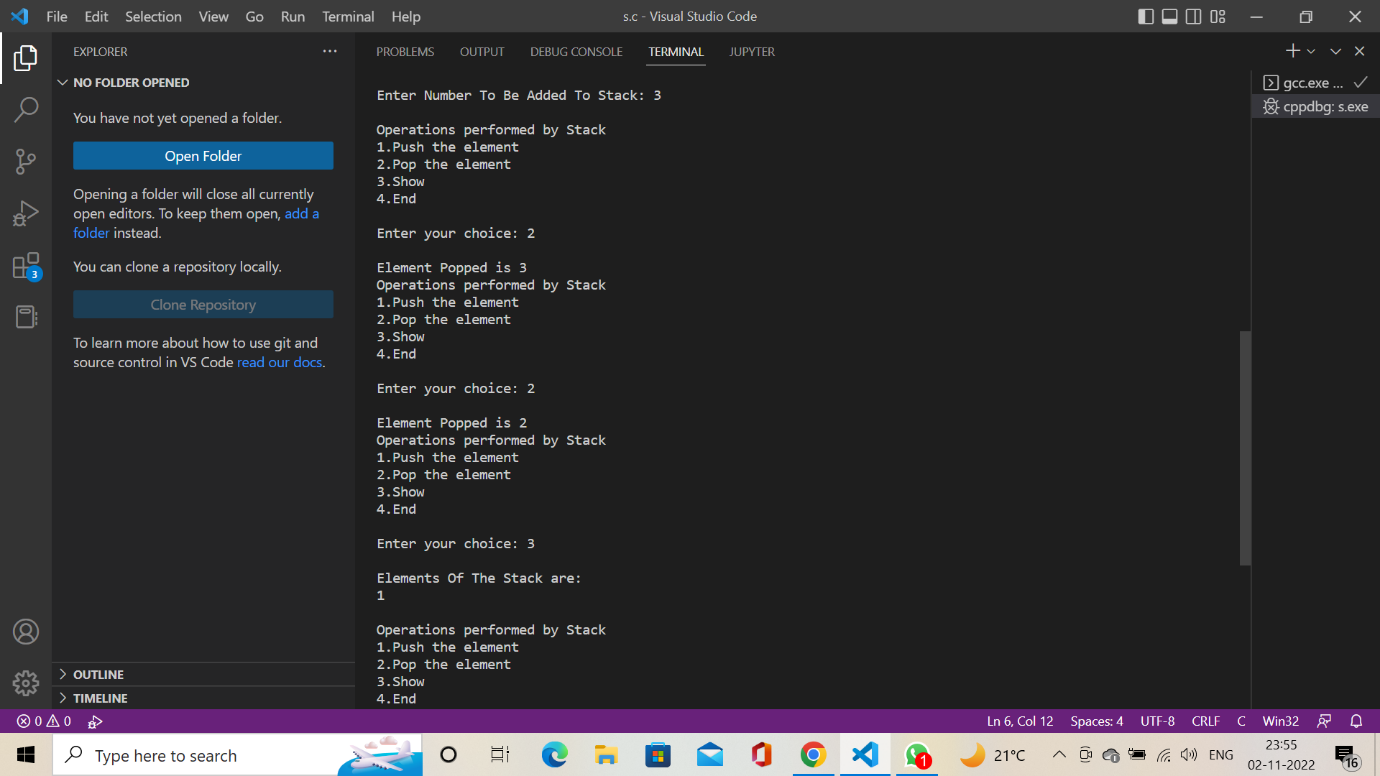
        }

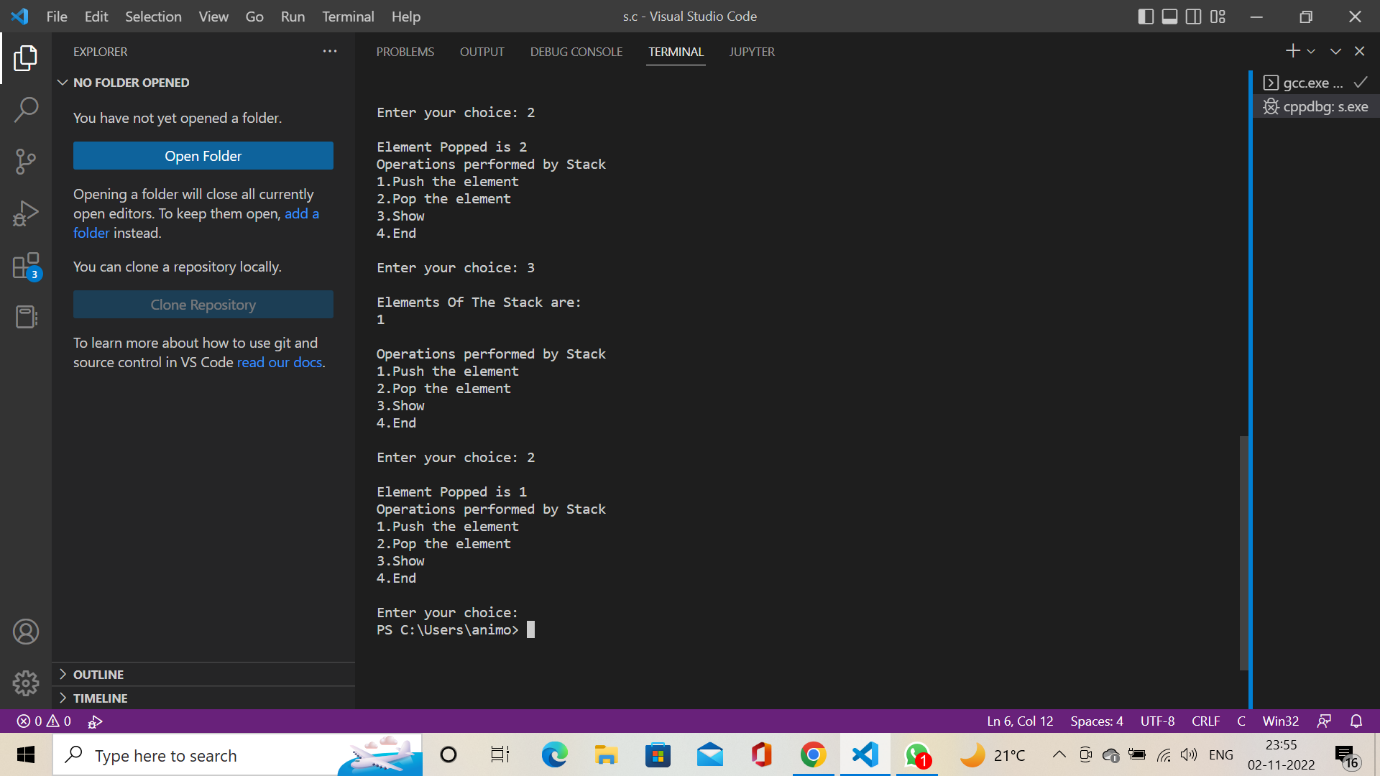
    }

}

**Output:**







**Exp.2.** Kindly use the stack data structure to reverse a number/string and if the number is a palindrome, print that number/string.

#include <stdio.h>

void numberStack(int arr[10]) {

    int remainder, num;

    int top = -1, copy, count=0, choice;

    printf("\nEnter the number : ");

    scanf("%d", &num);

    copy = num;

    //Pushing all the elements in the stack

    while (num != 0) {

        remainder = num % 10;

        arr[top] = remainder;

        num /= 10;

        top++;

    }

    num = copy;

    top--;

    while(num != 0) {

        remainder = num % 10;

        if(remainder == arr[top--]) {}

        else {

            count = 1;

        }

        num /= 10;

    }

    if(count == 1) {

        printf("\nThe number that you have entered is NOT a PALINDROME\n");

    }

    else {

        printf("\nThe number that you have entered is a PALINDROME\n");

    }

}

void stringStack(int arr[10]) {

    char str[20], i = 0, count = 0;

    printf("\nEnter the string : ");

    scanf("%s ",str);

        printf("entered -- %c", str[0]);

    //Pushing all the elements in the stack

    while (str[i] != '\0') {

        printf("entered -- %c", str[i]);

        arr[i] = str[i++];

    }

    i = 0;

    while(str[i] != '\0') {

        if(arr[i] == str[i]) {}

        else {

            count = 1;

        }

    }

    if(count == 1) {

        printf("\nThe string that you have entered is NOT a PALINDROME\n");

    }

    else {

        printf("\nThe string that you have entered is a PALINDROME\n");

    }

}

int main() {

    int top = -1, copy, count=0, choice;

    int num, remainder;

    int stackarr[10];

    printf("\n1. String\n2. Number\n");

    printf("\nChoose One : ");

    scanf("%d", &choice);

    switch(choice) {

        case 1:

            stringStack(stackarr);

            break;

        case 2:

            numberStack(stackarr);

            break;

        default:

            break;

    }

    return 0;

}

**Output:**

