Laboratory Component 3:

Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)

a. Push an Element on to Stack
b. Pop an Element from Stack
c. Demonstrate how Stack can be used to check Palindrome
d. Demonstrate Overflow and Underflow situations on Stack
e. Display the status of Stack
f. Exit
Support the program with appropriate functions for each of the above operations.

```
#include<stdio.h>
#include<stdlib.h>
#define MAXSIZE 5

int s[MAXSIZE];
int top = -1;

void push();
void pop();
void palindrome();
void display();

void main()
{
        int choice;
        while(1)
        {
                printf("\n\n\n\n~~~~~~Menu~~~~~~ : ");
                printf("\n=>1.Push an Element to Stack and Overflow demo ");
                printf("\n=>2.Pop an Element from Stack and Underflow demo");
                printf("\n=>3.Palindrome demo ");
                printf("\n=>4.Display ");
                printf("\n=>5.Exit");
                printf("\nEnter your choice: ");
                scanf("%d", &choice);
                switch(choice)
                {
                        case 1:  push();
                                 break;
                        case 2:  pop();
                                 break;
                        case 3: palindrome();
                                break;
                        case 4: display();
                                break;
                        case 5:  exit(0);
                                 break;
                        default: printf("\n Please enter valid choice ") ;
                                  break;
```

```
                }
        }
}


void push()
{
          if(top == MAXSIZE -1)
        {
                printf("\n~~~~Stack overflow~~~~");
                return;
        }
        else
        {
        int item;
        printf("\n Enter an element to be pushed: ");
        scanf("%d", &item);
        s[++top] = item;
        }
 }



void pop()
{
       if(top == -1)
           printf("\n~~~~Stack underflow~~~~");
       else
          printf("\n Element popped is: %d", s[top--]);
 }



void display()
{
        int i;
        if(top == -1)
        {
                printf("\n~~~~Stack is empty~~~~");
                return;
        }
        printf("\nStack elements are:\n ");
        for(i=top; i>=0 ; i--)
                printf("| %d |\n", s[i]);
}
```

```
void palindrome()
{
	int flag=1,i;
	printf("\nStack content are:\n");
	for(i=top; i>=0 ; i--)
		printf("| %d |\n", s[i]);

	printf("\nReverse of stack content are:\n");
	for(i=0; i<=top; i++)
		printf("| %d |\n", s[i]);

	for(i=0; i<=top/2; i++)
	{
		if( s[i] != s[top-i] )
		{
			flag = 0;
			break;
		}
	}
	if(flag == 1)
	{
		printf("\nIt is palindrome number");
	}
	else
	{
		printf("\nIt is not a palindrome number");
	}
}
```