**3. Design, Develop and Implement 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<stdlib.h>

#include<stdio.h>

#include<string.h>

#define max\_size 5

int stack[max\_size],top=-1;

void push();

void pop();

void display();

void pali();

int main()

{

int choice;

while(choice)

{

//printf("\n");

printf("\n\n--------STACK OPERATIONS-----------\n");

printf("1.Push\n");

printf("2.Pop\n");

printf("3.Palindrome\n");

printf("4.Display\n");

printf("5.Exit\n");

printf("-----------------------");

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

scanf("%d",&choice);

switch(choice)

{

case 1: push();

break;

case 2: pop();

break;

case 3: pali();

break;

case 4: display();

break;

case 5: exit(0);

break;

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

break;

}

}

return 0;

}

void push() //Inserting element into the stack

{

int item,n;

if(top==(max\_size-1))

{

printf("\nStack Overflow:");

}

else

{

printf("Enter the element to be inserted:\t");

scanf("%d",&item);

top=top+1;

stack[top]=item;

}

}

void pop() //deleting an element from the stack

{

int item;

if(top==-1)

{

printf("Stack Underflow:");

}

else

{

item=stack[top];

top=top-1;

printf("\nThe popped element: %d\t",item);

}

}

void pali()

{

int digit,j,k,len=top+1,flag=0,ind=0;

int num[len],rev[len],i=0;

while(top!=-1)

{

digit= stack[top];

num[i]=digit;

top--;

i++;

}

for(j=0;j<len;j++)

{

printf("Numbers= %d\n",num[j]);

}

printf("reverse operation : \n");

for(k=len-1;k>=0;k--)

{

rev[k]=num[ind];

ind++;

}

printf("reverse array : ");

for(k=0;k<len;k++)

{

printf("%d\n",rev[k]);

}

printf("check for palindrome :\n");

int length = 0;

for(i=0;i<len;i++)

{

if(num[i]==rev[i])

{

length = length+1;

}

}

if(length==len)

{

printf("It is a palindrome number\n");

}

else

{

printf("It is not a palindrome number\n");

}

top = len-1;

}

void display()

{

int i;

if(top==-1)

{

printf("\nStack is Empty:");

}

else

{

printf("\nThe stack elements are:\n" );

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

{

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

}

}

}

**Output:**

root:~/dslab # gedit stack.c

root:~/dslab # cc stack.c

root:~/dslab # ./a.out

--------STACK OPERATIONS-----------

1.Push

2.Pop

3.Palindrome

4.Display

5.Exit

-----------------------

Enter your choice: 1

Enter the element to be inserted: 7

Enter your choice: 1

Enter the element to be inserted: 4

Enter your choice: 1

Enter the element to be inserted:7

Enter your choice: 1

Enter the element to be inserted: 5

Enter your choice: 2

The popped element: 5

Enter your choice: 4

The stack elements are:

7

4

Enter your choice: 3

Numbers= 7

Numbers= 4

Numbers= 7

Reverse operation :

Reverse array :

7

4

7

Check for palindrome :

It is palindrome number