1. **PROGRAM TO IMPLEMENT STACK IN ARRAY**

#include<stdio.h>

#include<conio.h>

#include<process.h>

int stack[10];

static int top=-1;

void push()

{

 if(top==9)

 {

 printf("\nOVERFLOW");

 exit(0);

 }

 else

 {

 top++;

 printf("\nenter the element: ");

 scanf("%d",&stack[top]);

 }

}

void pop()

{

 if(top==-1)

 {

 printf("UNDERFLOW");

 exit(0);

 }

 else

 {

 printf("The deleted element is: %d ",stack[top]);

 stack[top]=NULL;

 top--;

 }

}

main()

{

clrscr();

int ch;

int cha=1;

int chaa=1;

printf("\n1.Push an element\n2.Pop an element\n3.Display\n4.Exit");

while(1)

{

printf("\nEnter your choice from MENU:");

scanf("%d",&ch);

switch(ch)

{

case 1:while(cha==1)

 {

 push();

 printf("do you wish to continue?(1/0):");

 scanf("%d",&cha);

 }

 break;

case 2: while(chaa==1)

 {

 pop();

 printf("do you wish to continue?(1/0):");

 scanf("%d",&chaa);

 }

 break;

case 3: printf("\nThe STACK is:\n");

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

 {

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

 }

 break;

 case 4: exit(0);

}

getch();

}

}

OUTPUT:



1. **PROGRAM TO IMPLEMENT QUEUE IN ARRAY**

# include <stdio.h>

#include<conio.h>

#include<process.h>

# define SIZE 10

int arr[ SIZE ], front = -1, rear = -1, i ;

void enqueue() ;

void dequeue() ;

void display() ;

int main()

{

 int ch,ch1=1 ;

 int ch2=1;

 clrscr();

 printf("\n\tMENU\n[1].ENQUEUE\n[2].DEQUEUE\n[3].Display\n[4].Exit\n") ;

 do

 {

 printf( "Enter your choice from MENU [1-4] : " ) ;

 scanf( "%d", &ch ) ;

 switch ( ch )

 {

 case 1 :while(ch1==1)

 {

 enqueue() ;

 printf("\nWant to enter more...(1/0)");

 scanf("%d",&ch1);

 }

 break ;

 case 2 :while(ch2==1)

 {

 dequeue() ;

 printf("\nWant to enter more...(1/0)");

 scanf("%d",&ch2);

 }

 break ;

 case 3 : display() ;

 break ;

 case 4 : exit(0);

 default : printf( "Invalid option\n" ) ;

 }

 }

 while ( ch != 4 ) ;

 return 0;

}

void enqueue()

{

 if ( rear == SIZE)

 {

 printf( "Queue is full (overflow)\n" ) ;

 return ;

 }

 rear++ ;

 printf( "Enter the element to ENQUEUE : " ) ;

 scanf( "%d", &arr[ rear ] ) ;

 if ( front == -1 )

 front++ ;

}

void dequeue()

{

 if ( front == -1 )

 {

 printf( "Queue is empty (underflow)\n" );

 return ;

 }

 printf( "The DEQUEUE element is : %d\n", arr[ front ] ) ;

 if ( front == rear )

 front = rear = -1 ;

 else

 front++ ;

}

void display()

{

 if ( front == -1 )

 {

 printf( "Queue is empty (underflow)\n" ) ;

 return ;

 }

 printf( "The elements in queue are : FRONT -> " ) ;

 for ( i = front ; i <= rear ; i++ )

 printf( "%d<-", arr[ i ] ) ;

 printf( "REAR\n" ) ;

}

OUTPUT:

