WEEK 5 LAB 5

Q.

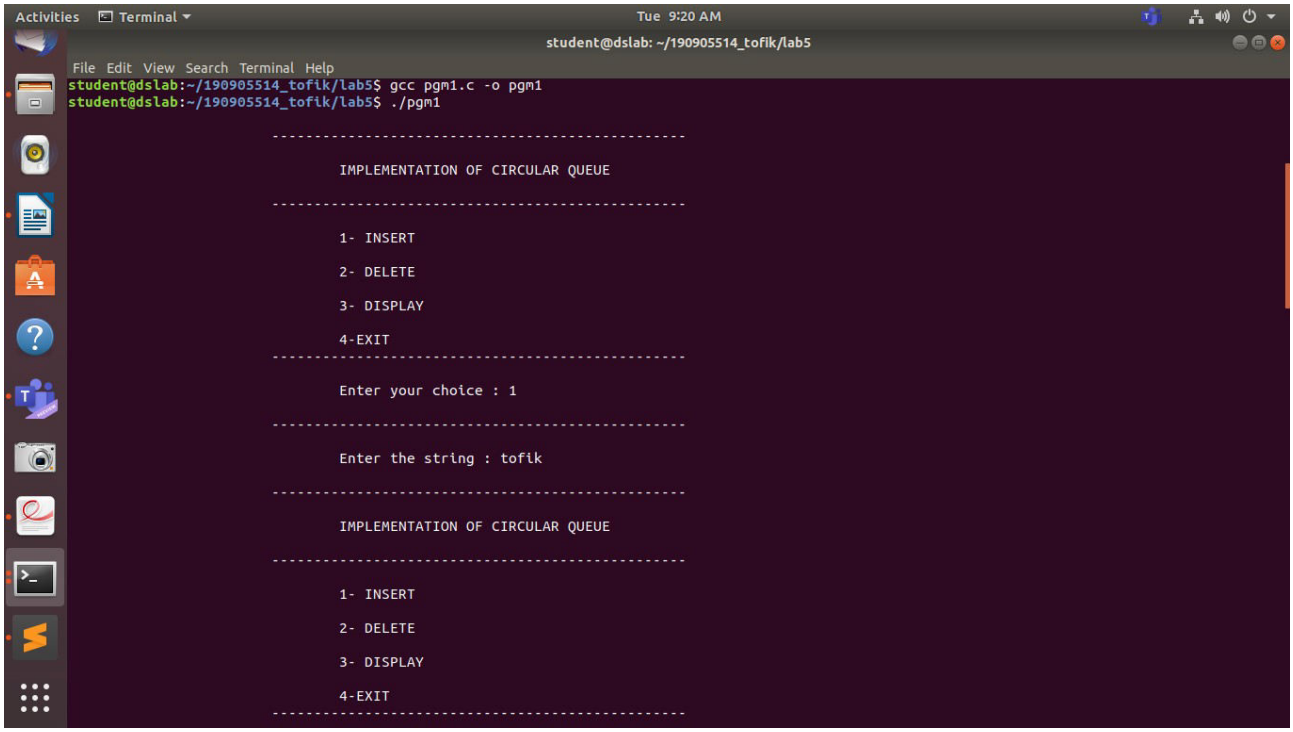
Implement a circular queue of Strings using structures. Include functions insertcq, deletecqand displaycq.

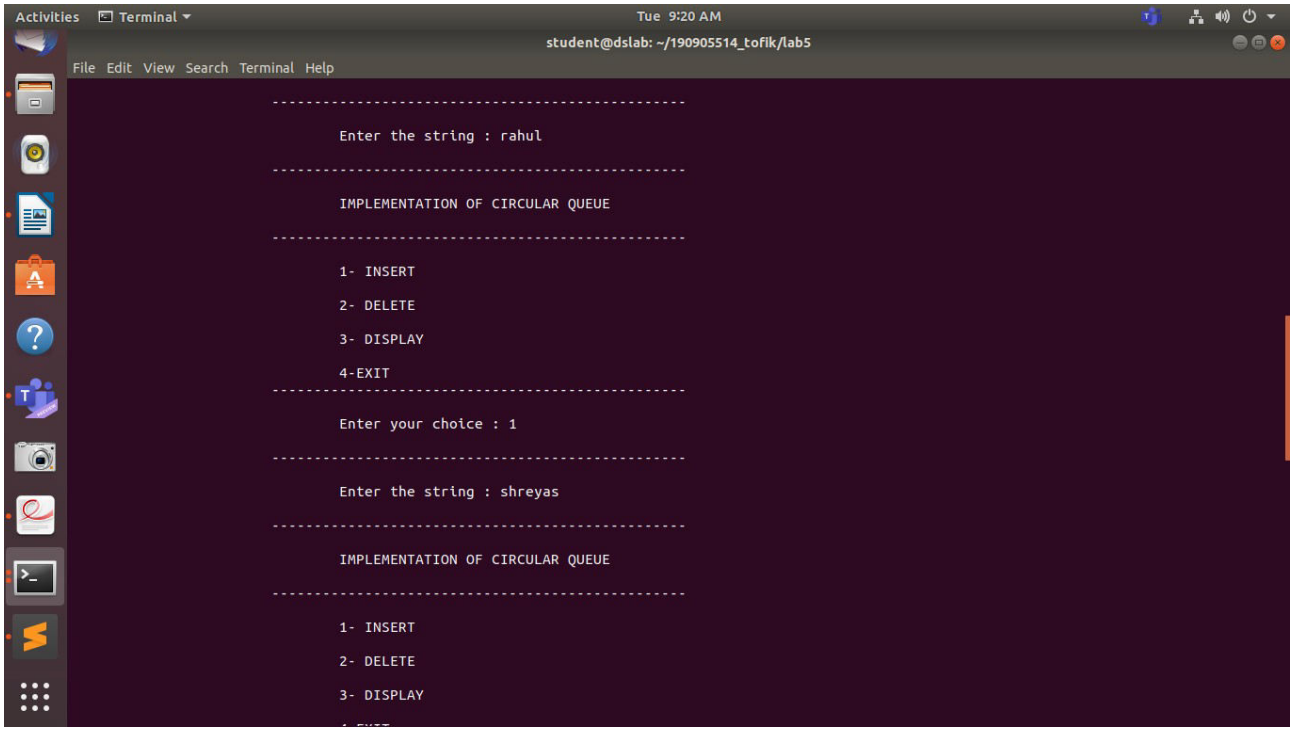
pgm1.c

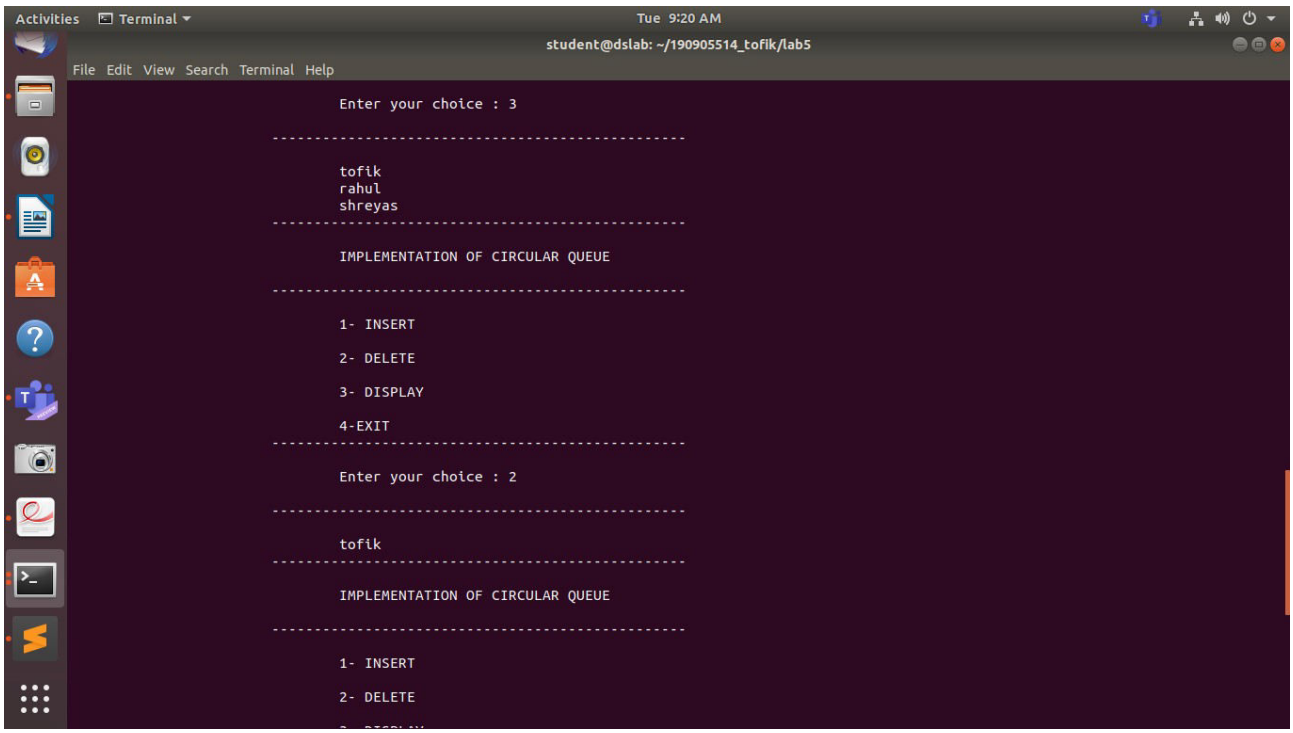
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define MAXIMUM 16
#define STRING 23
typedef struct
{
   char cqueue[MAXIMUM][STRING];
   int front;
   int rear;
} CQUEUE;

void insert(CQUEUE *queue,char a[])
{
   if((queue->rear+1)%(MAXIMUM+1)==queue->front)
   {
      printf("\n\t\t\t\t\tQueue is full :\n\n");
      return;
   }
   queue->rear=(queue->rear+1)%(MAXIMUM+1);
   strcpy(queue->cqueue[queue->rear],a);
}

void delete(CQUEUE *queue)
{
   if(queue->rear==queue->front)
   {
      printf("\n\t\t\t\t\tQueue is empty : \n\n");
      return;
   }
   char ch[STRING];
   queue->front=(queue->front+1)%(MAXIMUM+1);
   strcpy(ch,queue->cqueue[queue->front]);
   printf("\n\t\t\t\t\t%s",ch);
}

void display(CQUEUE *queue)
{
   if(queue->rear==queue->front)
   {
      printf(" \n\t\t\tQueue is empty : \n\n");
      return;
   }
   int myName =(queue->front+1)%(MAXIMUM+1);
```

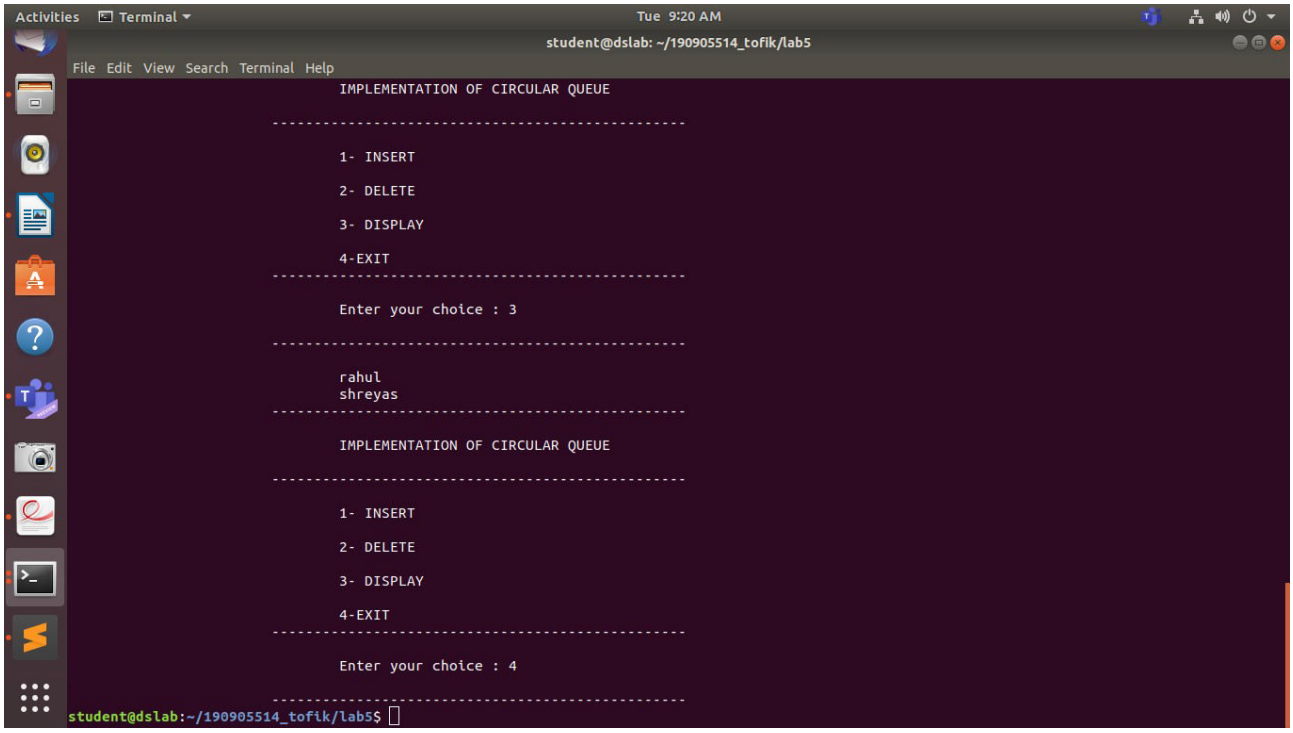
```
    for(; myName!=queue->rear; myName =(myName+1)%(MAXIMUM+1))
    {
        printf("\n\t\t\t\t\t%s  ",queue->cqueue[myName]);
    }
    printf("\n\t\t\t\t\t%s  ",queue->cqueue[myName]);
}

int main()
{


    char myName[MAXIMUM];
    CQUEUE *queue,queue1;
    queue=&queue1;
    queue->rear=queue->front=-1;
    while(1)
    {
        int choice;
        printf("\n\t\t\t\t----------------------------------------------------\n");
        printf("\n\t\t\t\t\tIMPLEMENTATION OF CIRCULAR QUEUE\n");
        printf("\n\t\t\t\t----------------------------------------------------\n");
        printf("\n\t\t\t\t\t1- INSERT\n");
        printf("\n\t\t\t\t\t2- DELETE \n");
        printf("\n\t\t\t\t\t3- DISPLAY\n");
        printf("\n\t\t\t\t\t4-EXIT");
        printf("\n\t\t\t\t----------------------------------------------------\n");
        printf("\n\t\t\t\t\tEnter your choice : ");
        scanf("%d",&choice);
        printf("\n\t\t\t\t----------------------------------------------------\n");
        switch(choice)
        {
        case 1:
            printf("\n\t\t\t\t\tEnter the string : ");

            scanf("%s",myName);

            insert(&queue1,myName);
            break;
        case 2:
            delete(&queue1);
            break;
        case 3:
            display(&queue1);
            break;
        case 4:
            exit(0);
            break;
        default :
            printf("\n\t\t\t\t\tInvalid Choice \n\n");
        }
    }
    return 0;
}
```

OUTPUT :


Activities
Terminal
Tue 9:20 AM
student@dslab: ~/190905514_tofik/lab5
File Edit View Search Terminal Help
student@dslab:~/190905514_tofik/lab5$ gcc pgm1.c -o pgm1
student@dslab:~/190905514_tofik/lab5$ ./pgm1
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT
Enter your choice : 1
Enter the string : tofik
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT


Activities
Terminal
Tue 9:20 AM
student@dslab: ~/190905514_tofik/lab5
File Edit View Search Terminal Help
Enter the string : rahul
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT
Enter your choice : 1
Enter the string : shreyas
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY


Activities
Terminal
Tue 9:20 AM
student@dslab: ~/190905514_tofik/lab5
File Edit View Search Terminal Help
Enter your choice : 3
tofik
rahul
shreyas
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT
Enter your choice : 2
tofik
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE


Activities
Terminal
Tue 9:20 AM
student@dslab: ~/190905514_tofik/lab5
File Edit View Search Terminal Help
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT
Enter your choice : 3
rahul
shreyas
IMPLEMENTATION OF CIRCULAR QUEUE
1- INSERT
2- DELETE
3- DISPLAY
4-EXIT
Enter your choice : 4
student@dslab:~/190905514_tofik/lab5$

Q.

Implement two circular queues of integers in a single array where first queue will run from 0 to N/2 and second queue will run from N/2+1 to N-1 where N is the size of the array.

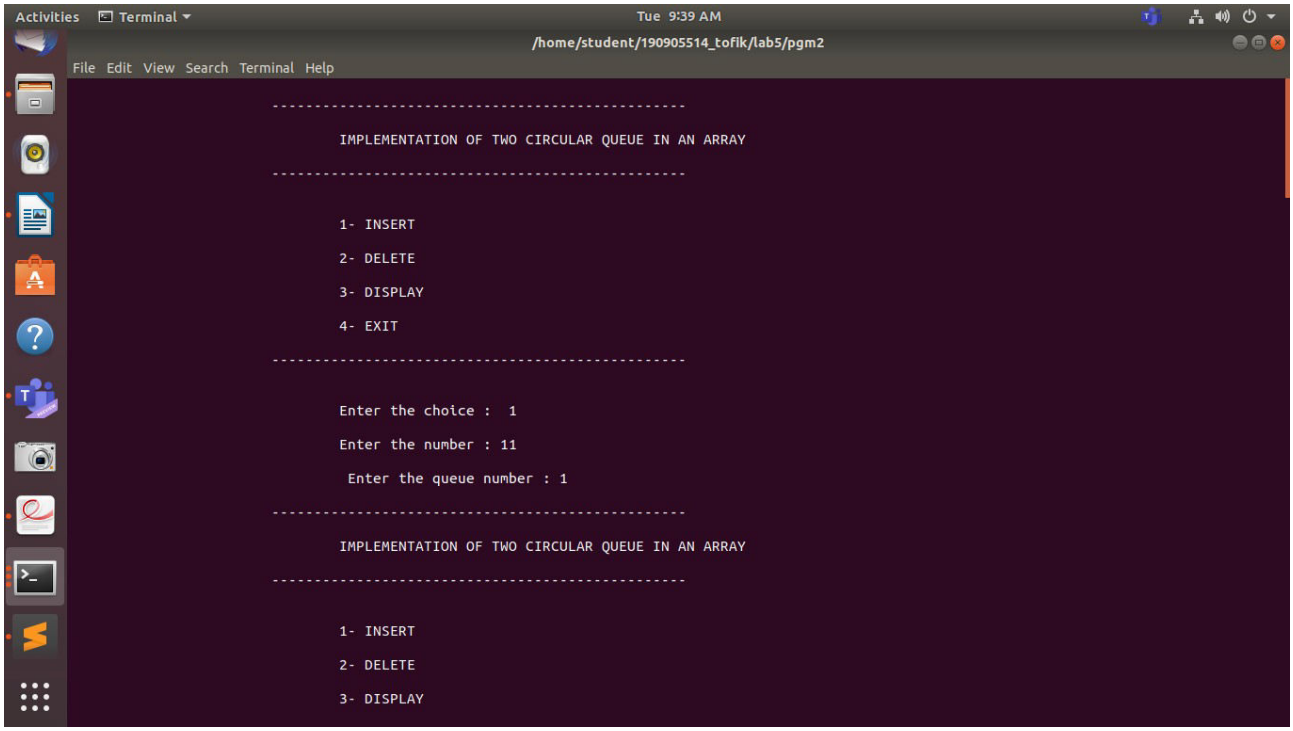
pgm2.c

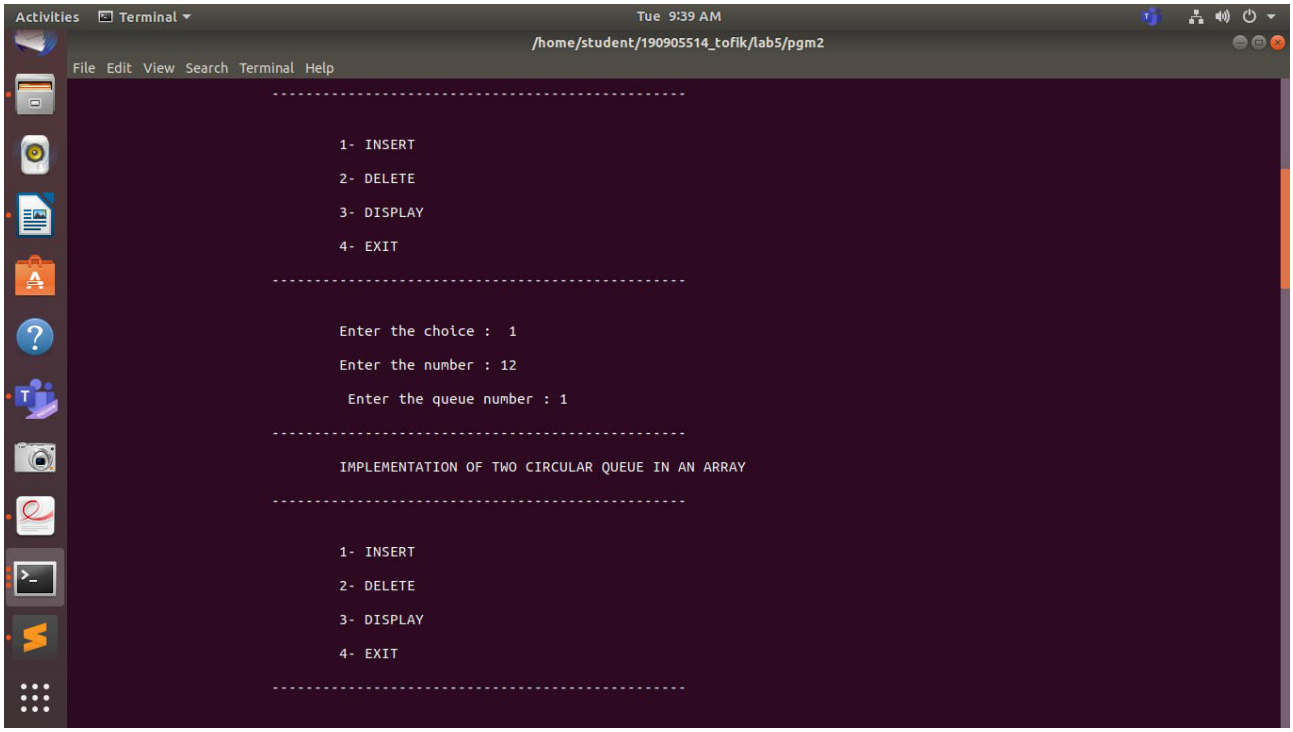
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define SIZE 20
#define n 2
int MAXIMUM = SIZE/n;
int queue[SIZE];
int front[n];
int rear[n];
void construct(){
	rear[0]=front[0]=0;
	rear[1]=front[1]=SIZE/n;

}
void insert(int i,int ch){
	if((rear[i]+1)%(MAXIMUM)+ (i*MAXIMUM)==front[i]){
		printf("\n\t\t\t\tQueue is empty: \n\n");
		return;
	}
	rear[i]=(rear[i]+1)%(MAXIMUM)+ (i*MAXIMUM);
	queue[rear[i]]=ch;
	//printf("\nfront and rear %d and %d\n",front[i],rear[i]);
}

int delete(int i){
	if(rear[i]==front[i]){
		printf("\n\t\t\t\tQueue is empty: \n\n");
		return -1;
	}
	front[i]=(front[i]+1)%(MAXIMUM)+ (i*MAXIMUM);
	return queue[front[i]];
}

void display(int i){
	if(rear[i]==front[i]){
		printf("\n\t\t\t\tQueue is empty : \n\n\n");
		return ;
	}
	int myName;
	printf("\n\t\t\t\tQueue is : \n\n  ");
	for(myName=(front[i]+1)%MAXIMUM+(i*MAXIMUM);
myName!=rear[i];
myName=((myName+1)%MAXIMUM+(i*MAXIMUM)))
```

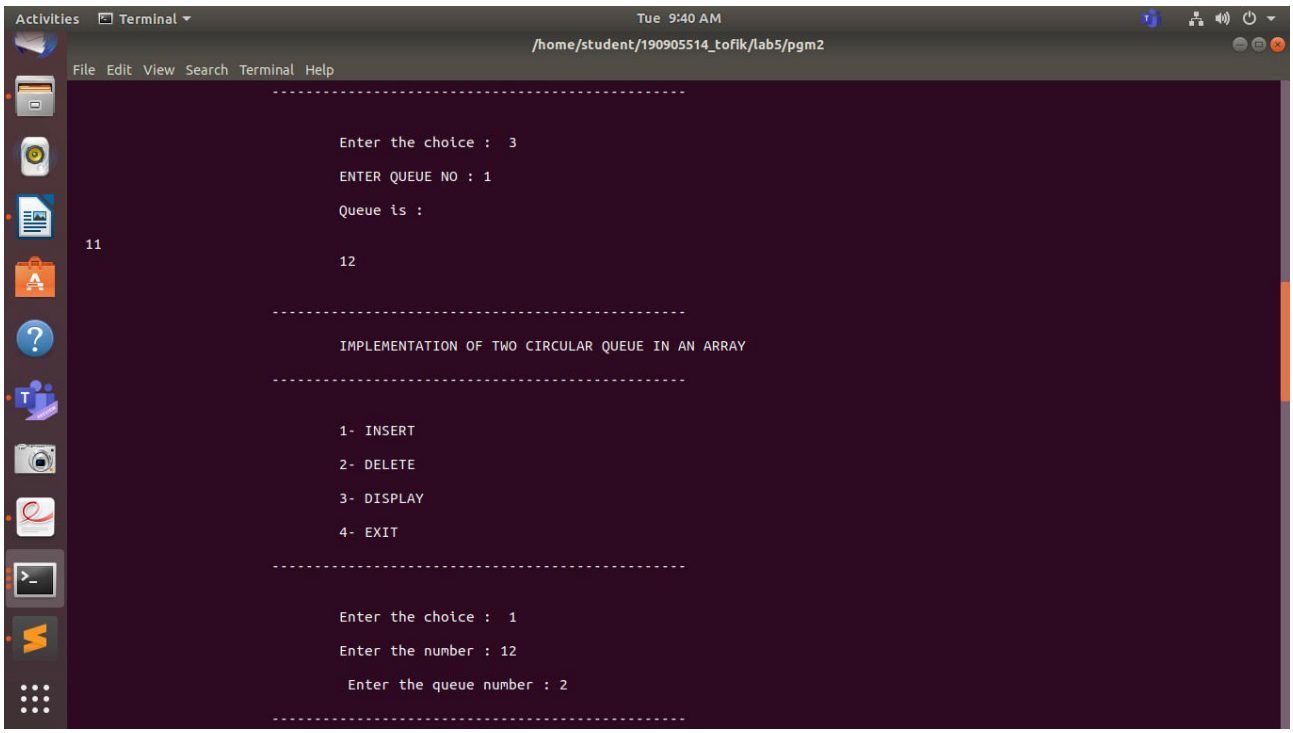
```
            printf("%d  ",queue[myName]);
            printf("\n\t\t\t\t%d  ",queue[myName]);
            printf("\n\n");
}
int main(){

            construct();

    int myName,i,j;
    while(1)
    {
        int choice;
          printf("\n\t\t\t----------------------------------------------\n");
        printf("\n\t\t\t\tIMPLEMENTATION OF TWO CIRCULAR QUEUE IN AN ARRAY\n");
        printf("\n\t\t\t----------------------------------------------\n\n");
                        printf("\n\t\t\t\t1- INSERT\n");
        printf("\n\t\t\t\t2- DELETE\n");
        printf("\n\t\t\t\t3- DISPLAY\n");
        printf("\n\t\t\t\t4- EXIT\n");
          printf("\n\t\t\t----------------------------------------------\n\n");
        printf("\n\t\t\t\tEnter the choice :  ");
        scanf("%d",&choice);
        switch(choice){
            case 1:
                        printf("\n\t\t\t\tEnter the number : ");
                        scanf("%d",&myName);
                        printf("\n\t\t\t\t Enter the queue number : ");
                                        scanf("%d",&i);
                                        insert(i-1,myName);
                                        break;
                                case 2:
                                        printf("\n\t\t\t\tEnter the queue number : ");
                                        scanf("%d",&i);
                                        j = delete(i-1);
                                        if(j!=-1)printf("\n\t\t\t\tELEMENT DELETED IS : %d\n",j);
                                        break;

                                case 3:
                                        printf("\n\t\t\t\tENTER QUEUE NO : ");
                                        scanf("%d",&i);
              display(i-1);
              break;

                                case 4 :
                                        printf("\n\t\t\t\texiting\n");
                                        exit(0);

                                default:
                                        printf("\n\t\t\t\twrong\n");
                                        break;



        }
```

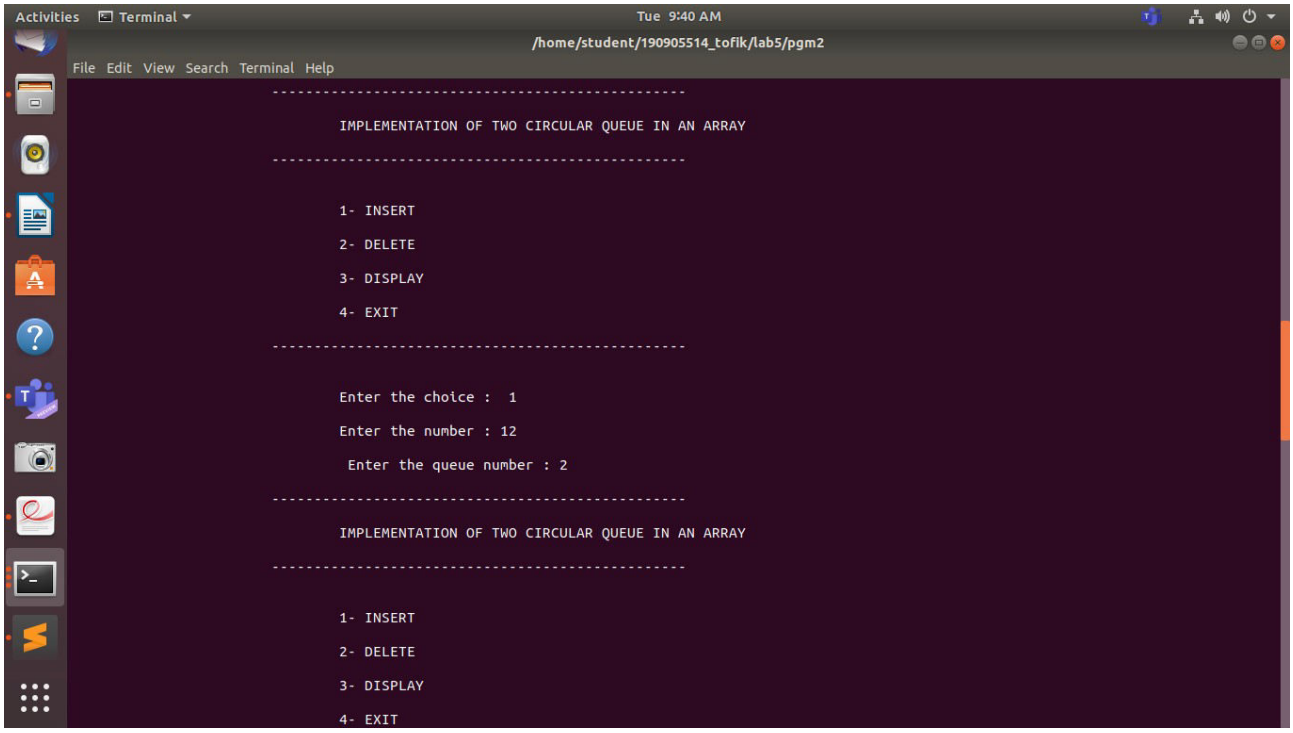
```
    }
    return 0;
}
```

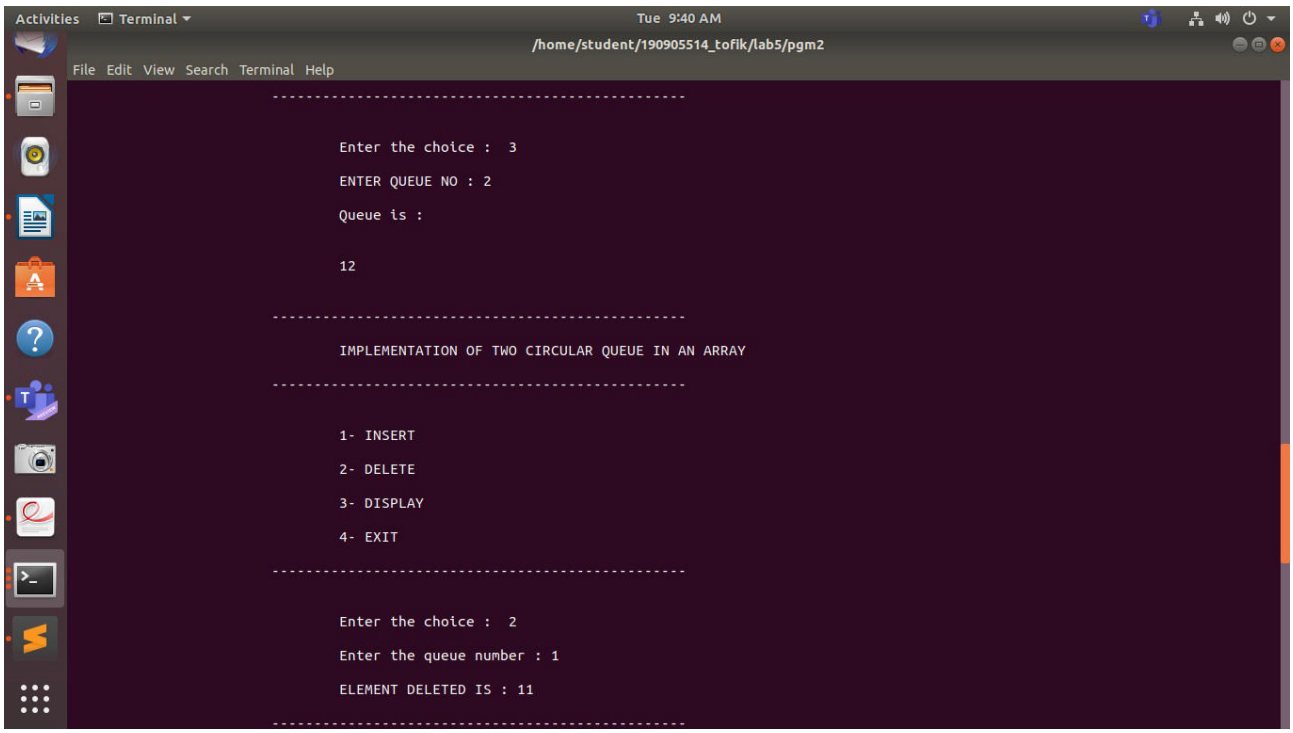
OUTPUT :

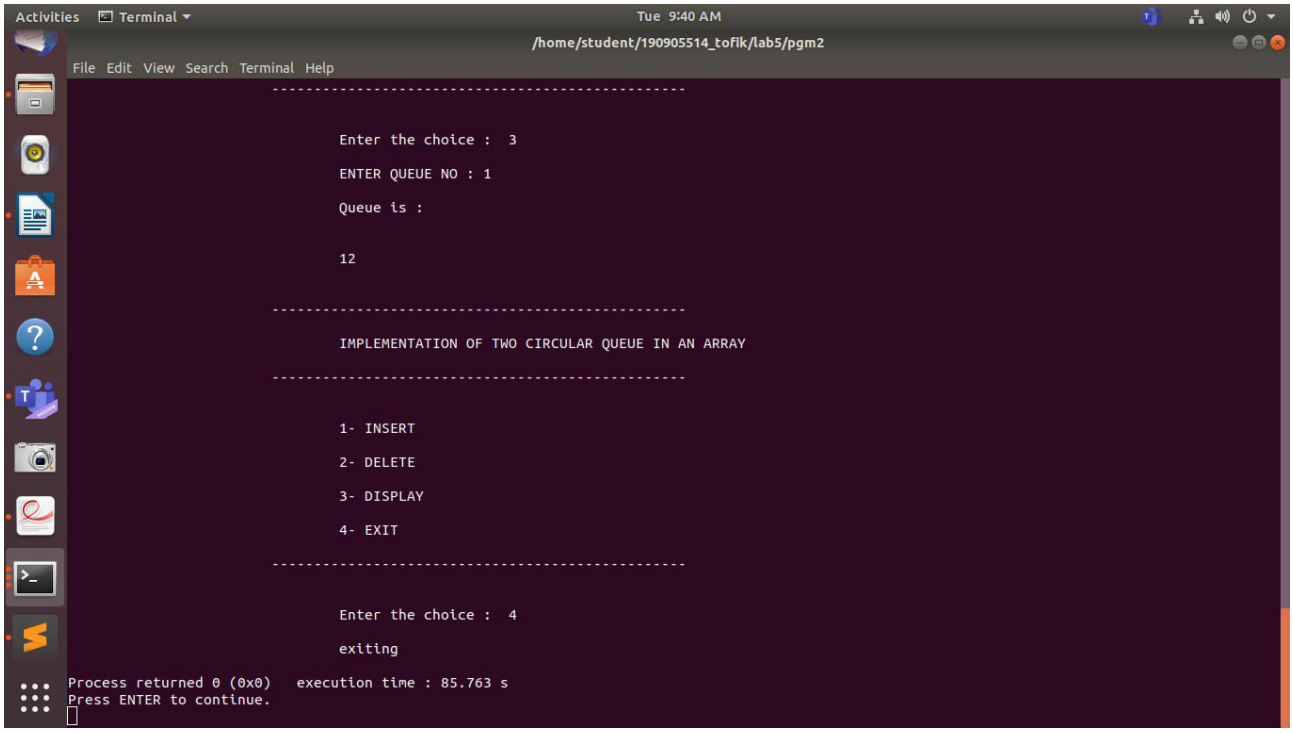





Activities
Terminal
Tue 9:40 AM
/home/student/190905514_tofik/lab5/pgm2
File Edit View Search Terminal Help
Enter the choice : 3
ENTER QUEUE NO : 1
Queue is :
11
12
IMPLEMENTATION OF TWO CIRCULAR QUEUE IN AN ARRAY
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT
Enter the choice : 1
Enter the number : 12
Enter the queue number : 2


Activities
Terminal
Tue 9:40 AM
/home/student/190905514_tofik/lab5/pgm2
File Edit View Search Terminal Help
IMPLEMENTATION OF TWO CIRCULAR QUEUE IN AN ARRAY
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT
Enter the choice : 1
Enter the number : 12
Enter the queue number : 2
IMPLEMENTATION OF TWO CIRCULAR QUEUE IN AN ARRAY
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT





Q.

Implement a queue with two stacks without transferring the elements of the second stack back to stack one. (use stack1 as an input stack and stack2 as an output stack).

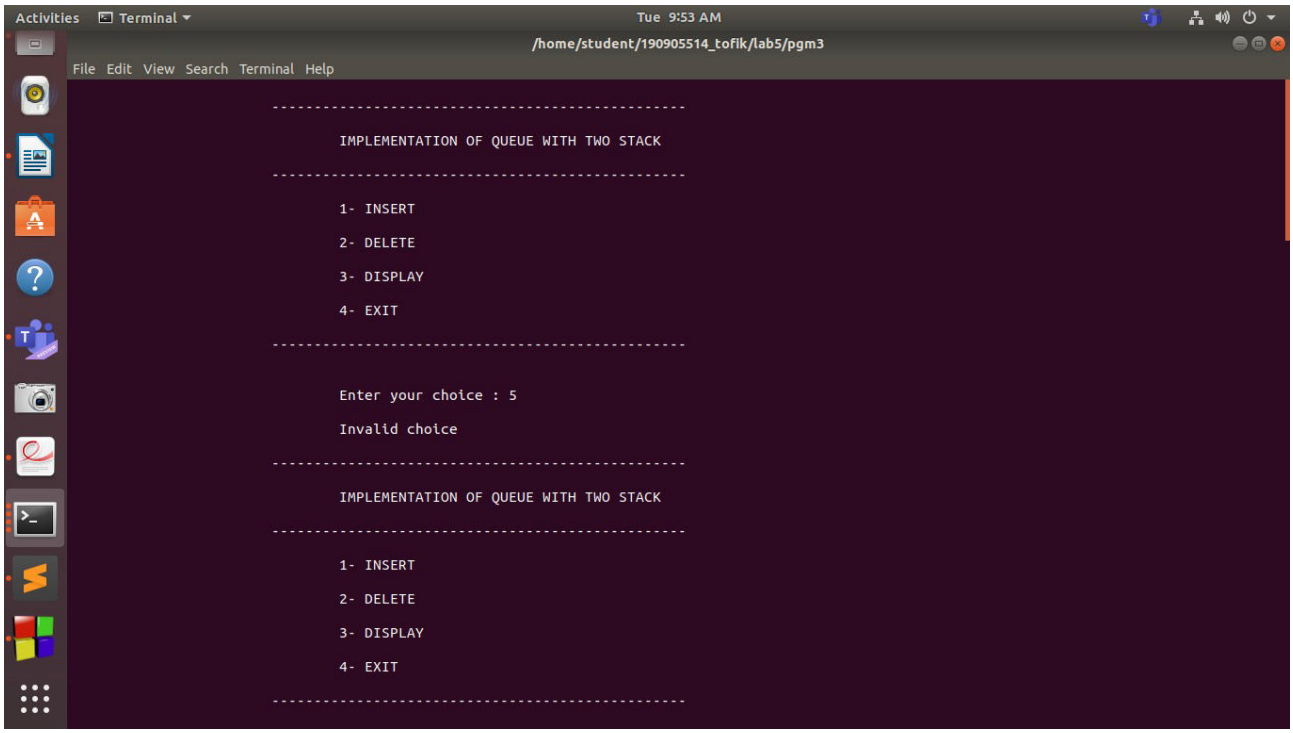
pgm3.c

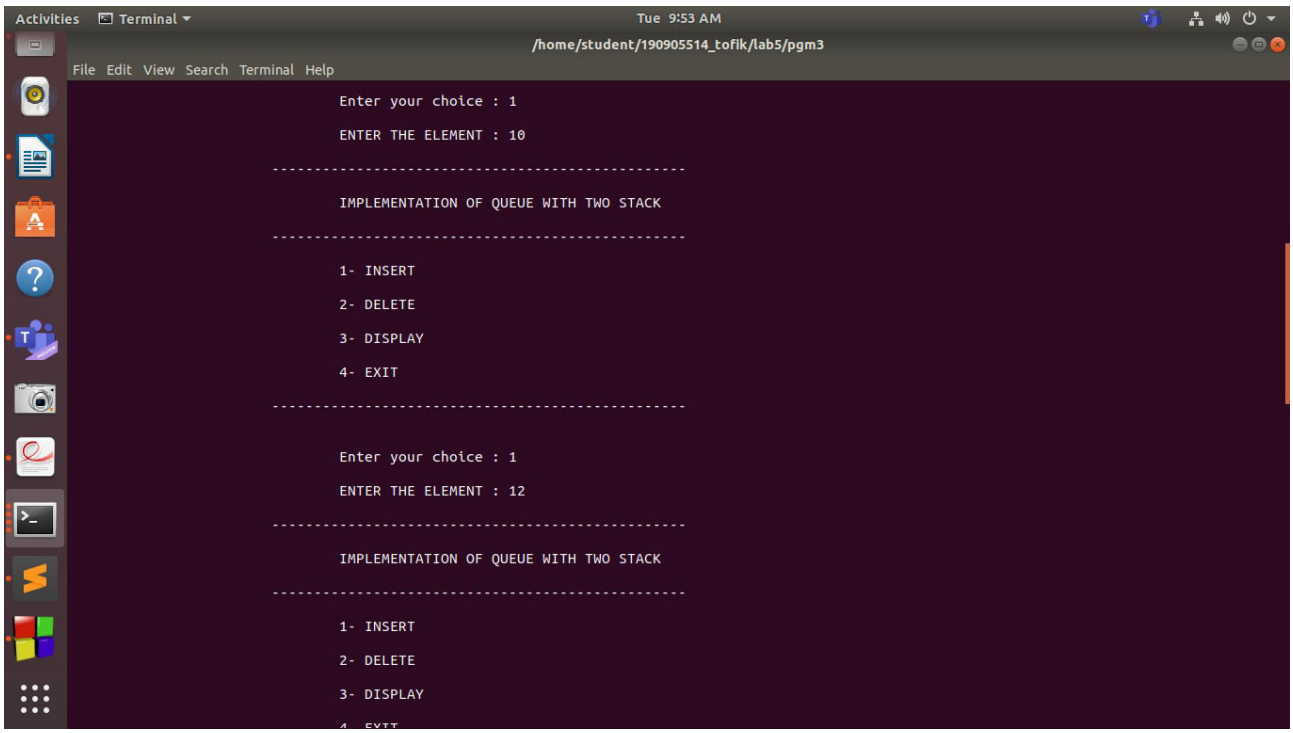
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
```

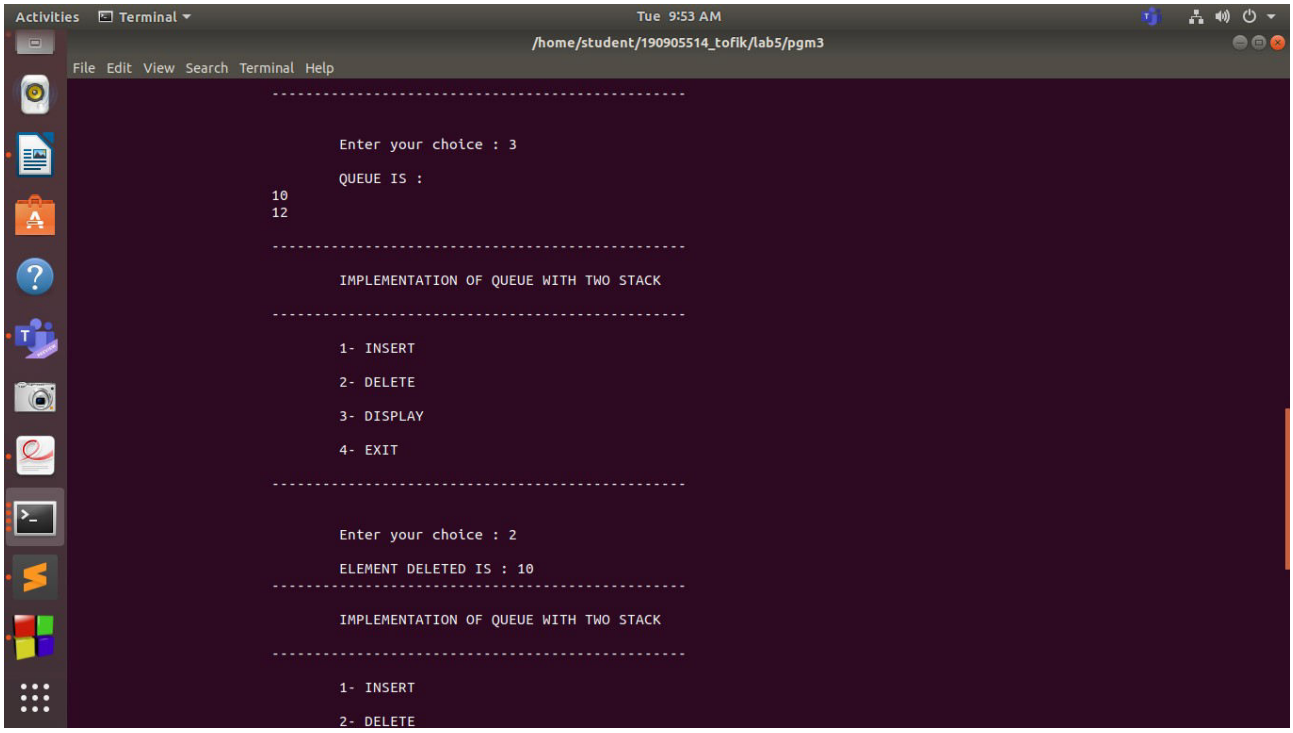
```
#define MAXSIZE 10
int input[MAXSIZE];
int output[MAXSIZE];
int top1=-1,top2=-1;

void insert(int i){
	if(top1==MAXSIZE-1){
		printf("\n\t\t\t\tQUEUE IS FULL\n");
		return;
	}
	input[++top1]=i;
}

int delete(){
	if(top1==-1 && top2==-1){
		printf("\n\t\t\t\tQUEUE IS EMPTY : \n");
		return -1;
	}
	if(top2==-1){
		while(top1!=-1){
			output[++top2]=input[top1--];
		}
	}
	return output[top2--];
}
void display(){
	if(top1==-1 && top2==-1){
		printf("\n\t\t\t\tQUEUE IS EMPTY : \n");
		return ;
	}
	int myName = top2;
	printf("\n\t\t\t\tQUEUE IS :  ");
	for(;myName>-1;myName--)
		printf("\n\t\t\t%d  ",output[myName]);
	for(myName=0;myName<=top1;myName++)
		printf("\n\t\t\t%d  ",input[myName]);
	printf("\n");
}

int main(){

   int x,i,j;
   while(1)
   {
      int choice;
      printf("\n\t\t\t----------------------------------------------\n");
      printf("\n\t\t\t\tIMPLEMENTATION OF QUEUE WITH TWO STACK \n");
      printf("\n\t\t\t----------------------------------------------\n");
      printf("\n\t\t\t\t1- INSERT\n");
      printf("\n\t\t\t\t2- DELETE\n");
      printf("\n\t\t\t\t3- DISPLAY\n");
      printf("\n\t\t\t\t4- EXIT\n");
      printf("\n\t\t\t----------------------------------------------\n\n");
      printf("\n\t\t\t\tEnter your choice : ");
```

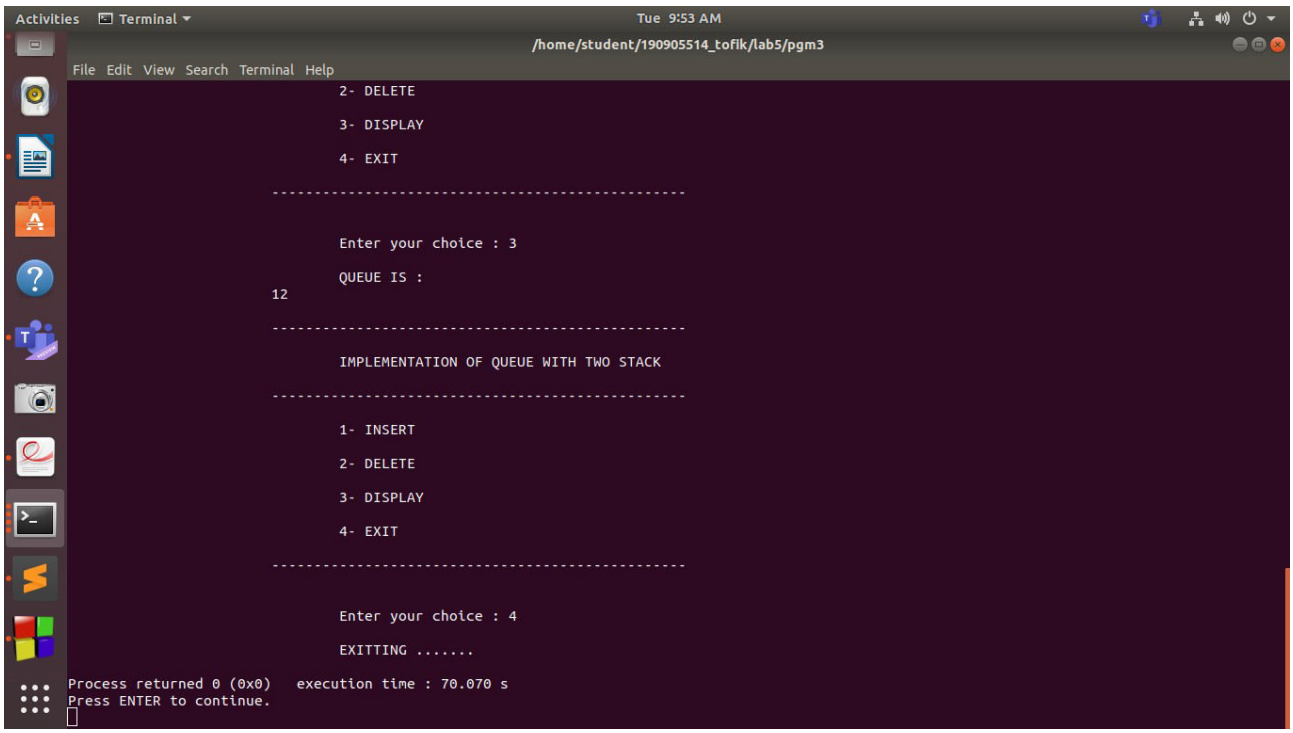
```
        scanf("%d",&choice);
        switch(choice){
            case 1:
                    printf("\n\t\t\t\tENTER THE ELEMENT : ");
                    scanf("%d",&x);
                                insert(x);
                                break;
                        case 2:
                                j = delete();
                                if(j!=-1)printf("\n\t\t\t\tELEMENT DELETED IS : %d",j);
                                break;

                        case 3:
                display();
                break;

                        case 4 :
                                printf("\n\t\t\t\tEXITTING .....\n");
                                exit(0);

                        default:
                                printf("\n\t\t\t\tInvalid choice\n");
                                break;



            }
        }
        return 0;
}
```

OUTPUT :




Activities
Terminal
Tue 9:53 AM
/home/student/190905514_tofik/lab5/pgm3
File Edit View Search Terminal Help
Enter your choice : 1
ENTER THE ELEMENT : 10
IMPLEMENTATION OF QUEUE WITH TWO STACK
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT
Enter your choice : 1
ENTER THE ELEMENT : 12
IMPLEMENTATION OF QUEUE WITH TWO STACK
1- INSERT
2- DELETE
3- DISPLAY


Activities
Terminal
Tue 9:53 AM
/home/student/190905514_tofik/lab5/pgm3
File Edit View Search Terminal Help
Enter your choice : 3
QUEUE IS :
10
12
IMPLEMENTATION OF QUEUE WITH TWO STACK
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT
Enter your choice : 2
ELEMENT DELETED IS : 10
IMPLEMENTATION OF QUEUE WITH TWO STACK
1- INSERT
2- DELETE


Activities
Terminal
Tue 9:53 AM
/home/student/190905514_tofik/lab5/pgm3
File Edit View Search Terminal Help
2- DELETE
3- DISPLAY
4- EXIT
Enter your choice : 3
QUEUE IS :
12
IMPLEMENTATION OF QUEUE WITH TWO STACK
1- INSERT
2- DELETE
3- DISPLAY
4- EXIT
Enter your choice : 4
EXITTING .....
Process returned 0 (0x0) execution time : 70.070 s
Press ENTER to continue.