**Institute of Information Technology**

**Jahangirnagar University**

### A picture containing drawing, art, sketch, design Description automatically generated

LAB REPORT No.- 12

2nd Year 1st Semester in ICT(Hons)

Course Code: ICT-2102

Course Title: Data Structure Lab

Submitted by: Submitted to:

Name: Abdullah Al Hossain Professor Dr. Jesmin Akhter,

Class roll:2044

Institute of

Institute of Information Technology Information Technology,

Jahangirnagar University Jahangirnagar University

**CODE 1:**

#include<bits/stdc++.h>

using namespace std;

#define SIZE 100

int arr[SIZE];

int Rear=-1;

int Front=-1;

main()

{

int i,insert\_item,delete\_item;

if(Front==-1)

Front=0;

cout<<"Element to be inserted:\n"<<endl;

for(i=0;i<5;i++){

cin>>insert\_item;

Rear=Rear+1;

arr[Rear]=insert\_item;

}

cout<<"Queue:\n"<<endl;

for(i=Front;i<=Rear;i++){

cout<<arr[i]<<" ";

cout<<endl;

}

delete\_item=arr[Front];

cout<<"\nElement deleted from the queue:"<<arr[Front]<<endl;

Front=Front+1;

if(Front==-1){

cout<<"Empty queue"<<endl;

}

else{

cout<<"Queue:"<<endl;

for(i=Front;i<=Rear;i++){

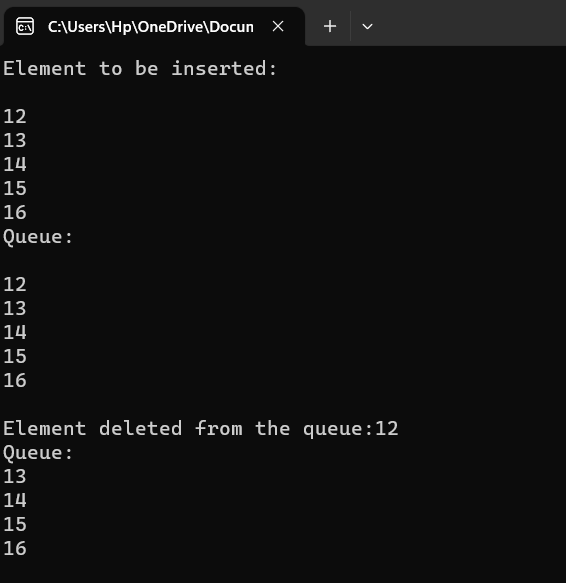
cout<<arr[i]<<endl;

}

}

}

**OUTPUT:**



**CODE 2::**

# include <stdio.h>

#include <cstdlib>

# define SIZE 100

void insert\_data();

void delete\_item();

void show();

int arr[SIZE];

int Rear=-1;

int Front=-1;

main(){

int k;

printf("1. Insertion Operation\n");

printf("2. Deletion Operation\n");

printf("3. Exit\n");

while (1)

{

printf("Enter your choice of operations: ");

scanf("%d", &k);

switch (k)

{

case 1:

insert\_data();

break;

case 2:

delete\_item();

break;

case 3:

exit(0);

break;

default:

printf("Incorrect choice \n");

break;

}

}

}

void insert\_data(){

int insert\_item;

if (Rear == SIZE-1)

{

printf("Overflow \n");

} else

{

if (Front == -1)

{

Front = 0;

}

printf("Element to be inserted in the Queue\n ");

scanf("%d", &insert\_item);

Rear=Rear+1;

arr[Rear]= insert\_item;

}

printf("\nAfter insertion: ");

show();

}

void delete\_item(){

if (Front == -1 || Front>Rear)

{

printf("Underflow \n");

return;

} else

{

printf("Element deleted from Queue: %d\n", arr[Front]);

Front=Front+1;

}

printf("\nAfter deletion: ");

show();

}

void show(){

if (Front == -1)

{

printf("Empty Queue\n ");

} else

{

printf("\nQueue: ");

for (int i = Front; i <= Rear; i++)

{

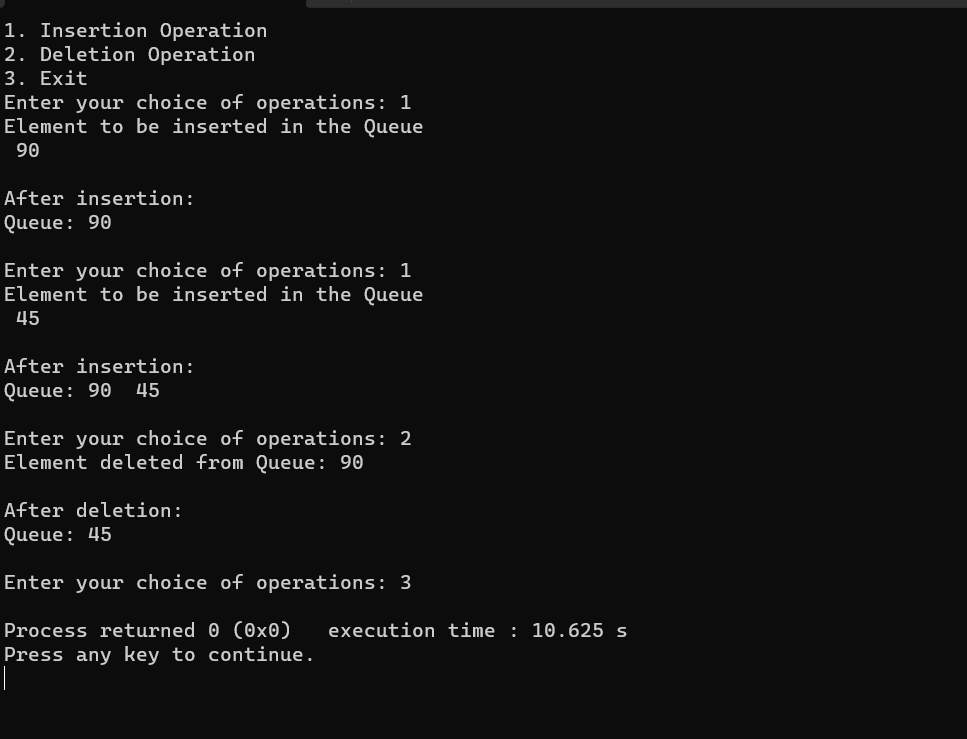
printf("%d ", arr[i]);

}

printf("\n\n");

}

}

OUTPUT:::: 

**CODE 3::::::::::**

#include <stdio.h>

#include <stdlib.h>

#define MAX 100

void insert\_data();

void delete\_item();

void show();

int arr[MAX];

int Rear = -1;

int Front = -1;

int main() {

int k;

printf("1. Insertion Operation\n");

printf("2. Deletion Operation\n");

printf("3. Exit\n");

while (1) {

printf("Enter your choice of operations: ");

scanf("%d", &k);

switch (k) {

case 1:

insert\_data();

break;

case 2:

delete\_item();

break;

case 3:

exit(0);

default:

printf("Incorrect choice\n");

break;

}

}

return 0;

}

void insert\_data() {

int insert\_item;

if ((Front == 0 && Rear == MAX - 1) || (Front == Rear + 1)) {

printf("Overflow\n");

return;

} else {

if (Front == -1) {

Front = 0;

Rear = 0;

} else {

if (Rear == MAX - 1)

Rear = 0;

else

Rear = Rear + 1;

}

printf("Element to be inserted in the queue:\n");

scanf("%d", &insert\_item);

arr[Rear] = insert\_item;

}

printf("\nAfter insertion:");

show();

}

void delete\_item() {

if (Front == -1) {

printf("Underflow\n");

return;

} else {

printf("Element deleted from the Queue: %d\n", arr[Front]);

if (Front == Rear) {

Front = -1;

Rear = -1;

} else {

if (Front == MAX - 1)

Front = 0;

else

Front = Front + 1;

}

}

printf("\nAfter deletion: ");

show();

}

void show() {

if (Front == -1)

printf("Empty Queue:\n");

else {

printf("\nQueue:");

if (Front <= Rear) {

for (int i = Front; i <= Rear; i++) {

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

}

} else {

for (int i = Front; i < MAX; i++) {

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

}

for (int i = 0; i <= Rear; i++) {

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

}

}

printf("\n");

}

}

**OUTPUT::::::::::**

