**PROGRAM OF CIRCULAR QUEUE:**

#include<stdio.h>

#include<stdlib.h>

#define SIZE 5

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

void insert(){

    int data;

    printf("Enter data to be inserted ");

    scanf("%d",&data);

    if((rear==SIZE-1 && front==0) || (front!=0 && (rear%SIZE)==front-1)){

        printf("Queue is Full\n \n");

        return;

    }

    else if (front==-1 && rear==-1)

    {

        rear=front=0;

        arr[rear]=data;

    }

    else if (front!=0 && rear==SIZE-1)

    {

        rear=0;

        arr[rear]=data;

    }

    else{

        rear++;

        arr[rear]=data;

    }

    printf("%d is inserted \n \n", data);

}

void delete(){

    if (front==-1){

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

        return;

    }

    else if (front==rear)

    {

        printf("%d is deleted \n \n",arr[front]);

        front=rear=-1;

    }

    else if (front==SIZE-1)

    {

        printf("%d is deleted \n \n",arr[front]);

        front=0;

    }

    else

    {

        printf("%d is deleted \n \n",arr[front]);

        front++;

    }

}

void display(){

    int i;

    if (front<=rear)

    {

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

        {

            printf("%d ",arr[i]);

        }

        printf("\n \n");

    }

    else{

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

        {

            printf("%d ",arr[i]);

        }

        for ( i = 0; i <= rear; i++)

        {

            printf("%d ",arr[i]);

        }

        printf("\n \n");

    }

}

int  main()

{

    int choice=-1;

    while(choice!=0){

        printf("Enter 1 to insert \nEnter 2 to delete \nEnter 3 to display \nEnter 0 to exit\n");

        scanf("%d",&choice);

        if(choice==1){

            insert();

        }

        else if (choice==2)

        {

            delete();

        }

        else if (choice==3)

        {

            display();

        }

        else if (choice==0)

        {

            break;

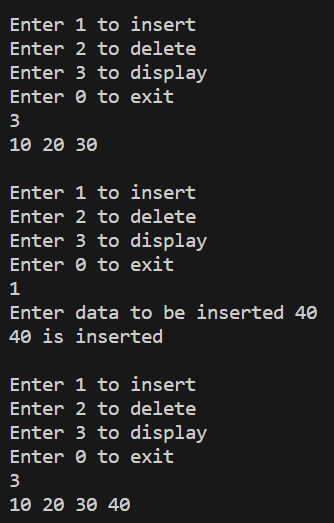
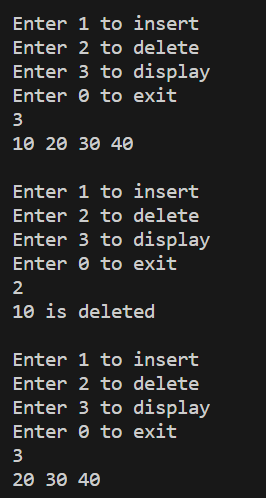
        }

    }

return 0;

}

**OUTPUT:**

** **

**PROGRAM OF deQUEUE:**

#include<stdio.h>

#include<stdlib.h>

#define SIZE 5

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

void insertFront(){

    int data;

    printf("Enter data to be inserted at the front: ");

    scanf("%d", &data);

    if ((rear == SIZE - 1 && front == 0) || (front != 0 && (rear % SIZE) == front - 1)){

        printf("Deque is Full\n \n");

        return;

    }

    else if (front == -1 && rear == -1){

        rear = front = 0;

        arr[rear] = data;

    }

    else if (front != 0 && rear == SIZE - 1){

        rear = 0;

        arr[rear] = data;

    }

    else{

        if (front == -1) front = 0;

        front = (front - 1 + SIZE) % SIZE; // Circular decrement

        arr[front] = data;

    }

    printf("%d is inserted at the front\n \n", data);

}

void insertRear(){

    int data;

    printf("Enter data to be inserted at the rear: ");

    scanf("%d", &data);

    if ((rear == SIZE - 1 && front == 0) || (front != 0 && (rear % SIZE) == front - 1)){

        printf("Deque is Full\n \n");

        return;

    }

    else if (front == -1 && rear == -1){

        rear = front = 0;

        arr[rear] = data;

    }

    else if (front != 0 && rear == SIZE - 1){

        rear = 0;

        arr[rear] = data;

    }

    else{

        if (rear == -1) rear = 0;

        rear = (rear + 1) % SIZE; // Circular increment

        arr[rear] = data;

    }

    printf("%d is inserted at the rear\n \n", data);

}

void deleteFront(){

    if (front == -1){

        printf("Deque is empty \n \n");

        return;

    }

    else if (front == rear){

        printf("%d is deleted \n \n", arr[front]);

        front = rear = -1;

    }

    else if (front == SIZE - 1){

        printf("%d is deleted \n \n", arr[front]);

        front = 0;

    }

    else{

        printf("%d is deleted \n \n", arr[front]);

        front = (front + 1) % SIZE; // Circular increment

    }

}

void deleteRear(){

    if (front == -1){

        printf("Deque is empty \n \n");

        return;

    }

    else if (front == rear){

        printf("%d is deleted \n \n", arr[rear]);

        front = rear = -1;

    }

    else if (rear == 0){

        printf("%d is deleted \n \n", arr[rear]);

        rear = SIZE - 1;

    }

    else{

        printf("%d is deleted \n \n", arr[rear]);

        rear = (rear - 1 + SIZE) % SIZE; // Circular decrement

    }

}

void display(){

    int i;

    if (front <= rear){

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

            printf("%d ", arr[i]);

        printf("\n \n");

    }

    else{

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

            printf("%d ", arr[i]);

        for (i = 0; i <= rear; i++)

            printf("%d ", arr[i]);

        printf("\n \n");

    }

}

int main(){

    int choice = -1;

    while (choice != 0){

        printf("Enter 1 to insert at the front\nEnter 2 to insert at the rear\nEnter 3 to delete from the front\nEnter 4 to delete from the rear\nEnter 5 to display\nEnter 0 to exit\n");

        scanf("%d", &choice);

        switch (choice){

            case 1:

                insertFront();

                break;

            case 2:

                insertRear();

                break;

            case 3:

                deleteFront();

                break;

            case 4:

                deleteRear();

                break;

            case 5:

                display();

                break;

            case 0:

                break;

            default:

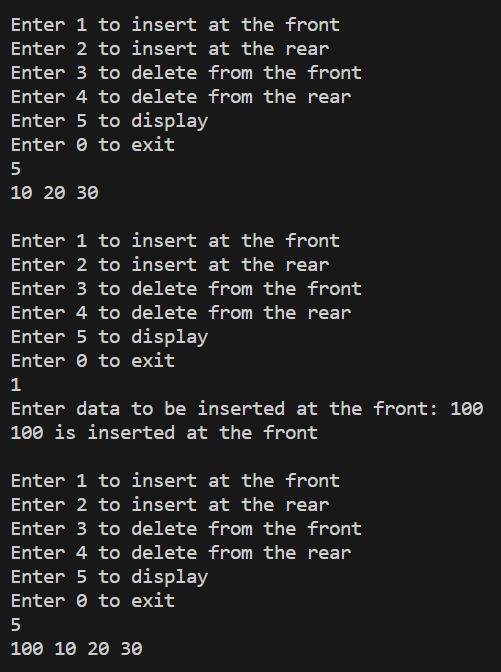
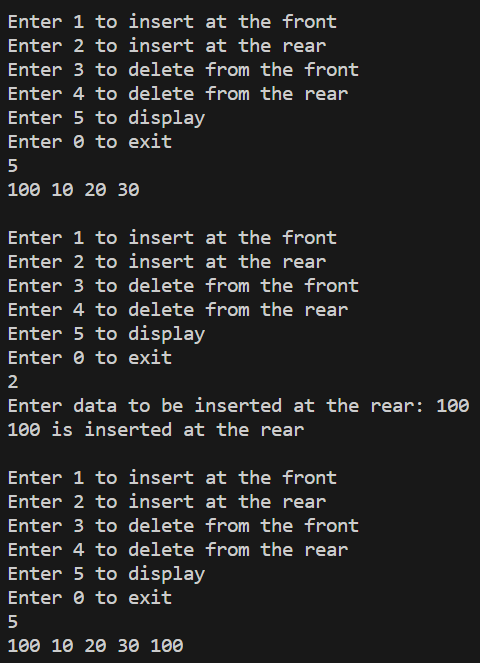
                printf("Invalid choice\n");

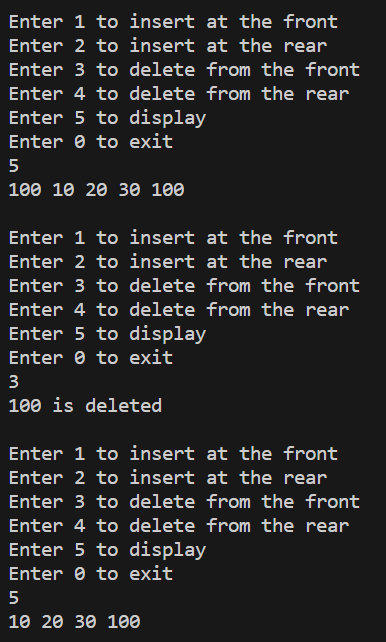
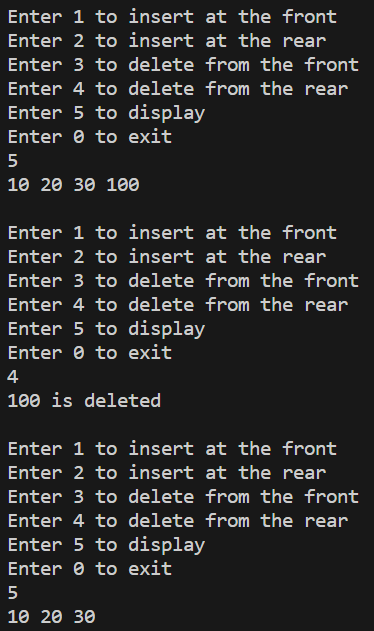
        }

    }

    return 0;

}

**OUTPUT:  
 **

** **