* Linear Queue

```
#define size 5
int queue[size], front = -1, rear = -1;
void enqueue();
int dequeue();
void display();

void enqueue(){
    int num;
    printf("Enter the number to insert:");
    scanf("%d", &num);
    if (rear == size-1){
        printf("overflow");
    } else if (front == -1 && rear == -1){
            front = 0;
            rear = 0;
    } else {
            rear++;
            queue[rear] = num;
    }

void display(){
    int i;
    printf("\n");
    if (front == -1 || front > rear)
            printf("Queue is empty");
    else{
        for(i = front; i <= rear; i++){
                printf("Element of Queue: %d", queue[i]);
        }
    };
```

```
int dequeue ( ){
    int value;
    if (front == -1 || front > rear){
        printf ("underflow");
    } else {
        value = queue [front];
        front ++;
        if (front > rear){
            front = -1;
            rear = -1;
        }
        return val;
    }
}
```

O/p:

```
**** MENU ****
Enter 1 : to insert
Enter 2 : to delete
Enter 3 : to display
Enter 4 : Exit

Enter your choice : 1
Enter the number to insert : 62

Enter your choice : 2
Deleted element is 62
Enter your choice : 3
 Element of queue : 42
 Element of queue : 27
Enter your choice : 4
        exit.
```

* Circular Queue

```
#include <stdio.h>

#define size 5
int queue[size], rear=-1, front=-1;

int isfull(){
    if((front==rear+1)||(front==0 && rear==size-1)){
        return 1;
    }
    return 0;
}

int isempty(){
    if(front==-1){
        return 1
    }
    return 0;
}

void enqueue(int element){
    if(isfull()){
        printf("Queue is full");
    }else{
        if(front==-1)
            front=0;
        rear=(rear+1)%size;
        queue[rear]=element;
        printf("Insertion successful : %d", element);
    }
}
```

```
int dequeue() {
    int value;
    if (isEmpty()) {
        printf("Queue is Empty");
        return -1;
    } else {
        value = queue[front];
        if (front == rear) {
            front = -1;
            rear = -1;
        } else {
            front = (front+1) % size;
        }
        return(value);
    }
}
```

Sur 8/1/24

```
void display() {
    int i;
    if (isEmpty())
        printf("Queue is Empty");
    else {
        printf("Front position: %d", front);
        for (i=front; i!=rear; i=(i+1)%size) {
            printf("%d", queue[i]);
        }
        printf("%d", queue[i]);
    }
}
```

o/p :

```
**** MENU ****
Enter 1 : to insert
Enter 2 : to delete
enter 3 : to display
Enter 4 : to exit
Enter your choice : 1
Enter the number to insert : 54
Enter your choice : 1
Enter the number to insert : 26
Enter your choice : 3
    Element in Queue : 54
    Element in Queue : 26
Enter your choice : 2
    Deleted element : 54
Enter your choice : 4
    Exit.
```