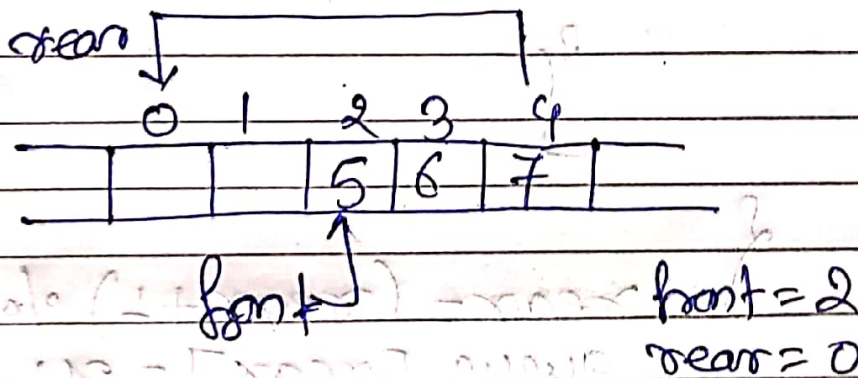
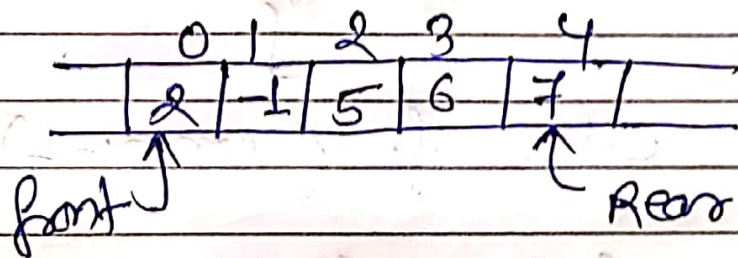


## Circular Queue

A Circular queue is a type of queue in which the last position is connected to the first position to make a circle.



Queue full condition :-

$$((\text{rear} + 1) \% \text{MAX}) == \text{front}$$

void enqueue (int x) {

if (front == -1 && rear == -1) {  
    front = rear = 0;  
    queue[rear] = x;  
}

else if ((rear+1) % MAX == front) {

    cout << "queue overflow" << endl;  
}

else {  
    rear = (rear+1) % N;  
    queue[rear] = x;  
}



void dequeue ( ) {

if ( front == -1 && rear == -1 ) {  
    cout << "queue, underflow" << endl;  
}

else if ( front == rear ) {  
    front = rear = -1;  
}

else {  
    cout << queue[front];  
    front = (front + 1) % MAX;  
}

void display ( ) {

int i = front;  
while ( i != rear ) {

    cout << queue[i] << " ";  
    i = (i + 1) % MAX;  
}

cout << queue[rear] << endl;  
}