**Aim:** Implement Circular Queue ADT using an array.

#include <stdio.h>

#define MAX\_SIZE 5

int queue[MAX\_SIZE];

int front =-1, rear =-1;

int isFull()

{

return (rear + 1) % MAX\_SIZE == front;

}

int isEmpty()

{

return front ==-1;

}

void enqueue(int data)

{

if (isFull()) { printf("Queue overflow\n"); return;

}

if (front ==-1) { front = 0;

}

rear = (rear + 1) % MAX\_SIZE;

queue[rear] = data;

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

}

int dequeue()

{

if (isEmpty()) { printf("Queue underflow\n"); return-1;

}

int data = queue[front];

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

}

else {

front = (front + 1) % MAX\_SIZE;

}

return data;

}

void display()

{

if (isEmpty()) {

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

}

printf("Queue elements: ");

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

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

i = (i + 1) % MAX\_SIZE;

}

printf("%d\n", queue[rear]);

}

int main()

{

enqueue(10); enqueue(20); enqueue(30);

display();

printf("Dequeued element: %d\n", dequeue());

display();

return 0;

}