#include <iostream>

class CircularQueue {

private:

static const int MAX\_ORDERS = 10;

int orders[MAX\_ORDERS];

int front, rear;

public:

CircularQueue() : front(-1), rear(-1) {}

// Function to check if the queue is empty

bool isEmpty() {

return front == -1 && rear == -1;

}

// Function to check if the queue is full

bool isFull() {

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

}

// Function to place a new order

void placeOrder(int orderNumber) {

if (isFull()) {

std::cout << "Maximum orders reached. Cannot place more orders." << std::endl;

return;

}

if (isEmpty()) {

front = rear = 0;

} else {

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

}

orders[rear] = orderNumber;

std::cout << "Order " << orderNumber << " placed successfully." << std::endl;

}

// Function to serve an order

void serveOrder() {

if (isEmpty()) {

std::cout << "No orders to serve. Queue is empty." << std::endl;

return;

}

int servedOrder = orders[front];

if (front == rear) {

// Last order in the queue

front = rear = -1;

} else {

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

}

std::cout << "Order " << servedOrder << " served successfully." << std::endl;

}

// Function to display the current orders in the queue

void displayOrders() {

if (isEmpty()) {

std::cout << "No orders in the queue." << std::endl;

return;

}

std::cout << "Current orders in the queue: ";

int current = front;

while (current != rear) {

std::cout << orders[current] << " ";

current = (current + 1) % MAX\_ORDERS;

}

// Print the last order

std::cout << orders[rear] << std::endl;

}

};

int main() {

CircularQueue pizzaQueue;

// Placing orders

pizzaQueue.placeOrder(1);

pizzaQueue.placeOrder(2);

pizzaQueue.placeOrder(3);

// Displaying the current state of the queue

pizzaQueue.displayOrders();

// Serving orders

pizzaQueue.serveOrder();

pizzaQueue.serveOrder();

// Displaying the updated state of the queue

pizzaQueue.displayOrders();

return 0;

}

**Output**

**Order 1 placed successfully.**

**Order 2 placed successfully.**

**Order 3 placed successfully.**

**Current orders in the queue: 1 2 3**

**Order 1 served successfully.**

**Order 2 served successfully.**

**Current orders in the queue: 3**