Pizza parlor accepting maximum M orders. Orders are served in first come first served basis. Order once placed cannot be cancelled. Write C++ program to simulate the system using circular queue using array.  
  
  
13. #include <iostream>

#include <string>

using namespace std;

class PizzaParlor {

private:

string \*queue; // Array to store orders (strings)

int front, rear, size;

public:

PizzaParlor(int M) {

size = M;

queue = new string[size];

front = rear = -1;

}

~PizzaParlor() {

delete[] queue;

}

// Check if the queue is full

bool isFull() {

return (front == 0 && rear == size - 1) || (front == rear + 1);

}

// Check if the queue is empty

bool isEmpty() {

return front == -1;

}

// Add an order to the queue

void placeOrder(string order) {

if (isFull()) {

cout << "Sorry, the pizza parlor is at full capacity. Cannot accept more orders." << endl;

return;

}

if (front == -1) { // If the queue is empty

front = rear = 0;

} else if (rear == size - 1 && front != 0) { // Wrap around if needed

rear = 0;

} else {

rear++;

}

queue[rear] = order;

cout << "Order placed: " << order << endl;

}

void serveOrder() {

if (isEmpty()) {

cout << "No orders to serve." << endl;

return;

}

cout << "Serving order: " << queue[front] << endl;

if (front == rear) { // Only one order in the queue

front = rear = -1;

} else if (front == size - 1) { // Wrap around if needed

front = 0;

} else {

front++;

}

}

void displayOrders() {

if (isEmpty()) {

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

return;

}

cout << "Current orders in the pizza parlor: ";

int i = front;

while (i != rear) {

cout << queue[i] << " ";

i = (i + 1) % size; // Circular increment

}

cout << queue[rear] << endl; // Print the last order

}

};

int main() {

int maxOrders;

cout << "Enter the maximum number of orders the pizza parlor can accept: ";

cin >> maxOrders;

PizzaParlor parlor(maxOrders);

int choice;

string order;

do {

cout << "\nPizza Parlor Menu:" << endl;

cout << "1. Place an Order" << endl;

cout << "2. Serve an Order" << endl;

cout << "3. Display Current Orders" << endl;

cout << "4. Exit" << endl;

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Enter the order (Pizza type): ";

cin >> order;

parlor.placeOrder(order); // Place a new order

break;

case 2:

parlor.serveOrder(); // Serve (remove) an order

break;

case 3:

parlor.displayOrders(); // Display all orders in the queue

break;

case 4:

cout << "Exiting the system." << endl;

break;

default:

cout << "Invalid choice. Please try again." << endl;

}

} while (choice != 4);

return 0;

}