Doctor's Office Patient Queue Management System.

(Implement queue using an array)

You are tasked with creating a menu-driven queue management system for a doctor's office. The system should allow the doctor's attendant to add patients to a queue, process them one by one, see who's next, and display the patient queue. Here's a detailed breakdown of the task:

Implement a simple queue data structure to manage the flow of patients waiting to see the doctor. Your program should first as how many seats are available in the waiting area, it means the max no of patients that can wait in the queue.

Create a menu-driven system with the following options:

1. **Add Patient to Queue:** The attendant can add a patient's name to the end of the queue.
2. **Send next patient to the doctor:** The attendant can process the next patient in the queue and send them to the doctor for examination. Remove the patient from the queue at this stage.
3. **See who's next:** The program should only display the name of the next patient.
4. **Display Patient Queue:** Display the current queue of patients waiting to see the doctor.

Sample Output:

```
Enter the number of seats in the waiting area:
5

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
1
Enter the name of the patient:
Fariba
Patient Fariba now waiting in the queue.

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
1
Enter the name of the patient:
Ali
Patient Ali now waiting in the queue.

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
1
Enter the name of the patient:
Saad
Patient Saad now waiting in the queue.

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
4
Patients in queue:
Fariba    Ali    Saad

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
2
Patient Fariba now with the doctor.
```

```
Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
3
Next Patient in queue: Ali

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
2
Patient Ali now with the doctor.

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
2
Patient Saad now with the doctor.

Enter an option:
1. Enter a patient in wating queue.
2. Send next patient to the doctor.
3. See who's next.
4. Display all patients in the queue.
2
No more patients.
```

Amusement Park Ticket Sales and Attraction Queue Management

(Implement queue using a Linked List)

You are tasked with creating a queue management system for an amusement park that includes ticket sales and multiple attractions. The system will use a simple queue data structure.

Ticket Sales Queue:

Implement a queue to manage the flow of visitors purchasing tickets at the amusement park entrance.

Visitors arrive at the ticket booth and join the ticket sales queue.

The ticket salesperson serves visitors one by one from the queue in a first-come, first-served manner. After selling tickets to a visitor, remove them from the queue, and add them to the relevant attraction queue.

Phase 2 - Attraction Queues:

Once visitors have purchased their tickets, they can access various attractions in the park.

Create separate queues for each attraction in the park (e.g., Roller Coaster, Round Wheel, and Motion ride). Each attraction has its own queue.

Implement a queue processing mechanism to manage the visitor flow to attractions. Process the queues one by one, starting with the Roller Coaster queue, then motion ride, and then round wheel's queue.

For each attraction queue, serve visitors one by one in a first-come, first-served manner.

After a visitor has enjoyed an attraction, remove them from the respective attraction queue.

Create a menu driven program for the ticket's sale's person with the following choices:

Enter visitor in the tickets purchase queue: Enters a visitor's name in the tickets purchase queue.

Sell a ticket: Sell a ticket to the next person from the ticket purchase queue. Based on the attraction's ticket, now the person should be in the waiting queue for that attraction.

Process all queues: Process and serve all the visitors (starting from the roller coaster's attraction, then motion ride, and the round wheel in first come first serve basis.

Sample Output:

```
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
1
Enter the visitor's name:
Fariba
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
1
Enter the visitor's name:
Asad
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
1
Enter the visitor's name:
Fawad
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
Now selling ticket to: Fariba...
Which attraction's ticket Fariba wants?
1. Roller Coaster
2. Motion Ride
3. Round Wheel
1
Roller coaster ticket sold to Fariba
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
Now selling ticket to: Asad...
Which attraction's ticket Asad wants?
1. Roller Coaster
2. Motion Ride
3. Round Wheel
2
Motion ride ticket sold to Asad
```

```
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
Now selling ticket to: Fawad...
Which attraction's ticket Fawad wants?
1. Roller Coaster
2. Motion Ride
3. Round Wheel
3
Round wheel ticket sold to Fawad
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
No more visitors in the queue
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
1
Enter the visitor's name:
Amir
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
1
Enter the visitor's name:
Alia
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
Now selling ticket to: Amir...
Which attraction's ticket Amir wants?
1. Roller Coaster
2. Motion Ride
3. Round Wheel
1
Roller coaster ticket sold to Amir
```

```
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
2
Now selling ticket to: Alia...
Which attraction's ticket Alia wants?
1. Roller Coaster
2. Motion Ride
3. Round Wheel
3
Round wheel ticket sold to Alia
Enter your choice:
1. Enter a visitor in tickets purchase queue.
2. Sell a ticket.
3. Process all queues.
3
Visitor Fariba now enjoying at the attraction roller coaster
Visitor Amir now enjoying at the attraction roller coaster
Visitor Asad now enjoying at the attraction motion ride
Visitor Fawad now enjoying at the attraction round wheel
Visitor Alia now enjoying at the attraction round wheel
```