

## Linear vs Circular Queue in Bus Ticketing

FEATURE	LINEAR QUEUE	CIRCULAR QUEUE
Memory Efficiency	Poor due to space unused after dequeue	Excellent yet it reuses freed space
Overflow Risk	High even if the space exists	Low because it wraps around
Implementation Simplicity	Easier	Slightly complex
Performance	Slower under heavy load	Faster and more consistent

**Conclusion:** Circular queues are more efficient for bus ticketing systems, especially when handling continuous passenger flow without wasting memory.

### Reflection: Why Queues Model Fairness in Rwanda

Queues embody fairness because they follow the **First-In-First-Out (FIFO)** principle. In Rwanda's banking halls, transport hubs, and telecom centers:

- **Time-based fairness:** Whoever arrives first is served first.
- **Transparency:** Everyone sees their position and progress.
- **No favoritism:** Reduces bias and corruption.
- **Reduces conflict:** Clear order prevents disputes.

Queues reflect **Ubuntu** values—respect, patience, and community order.

Would you like to simulate a circular queue in Python next, or explore how priority queues might handle emergencies in Rwandan hospitals?

### Reflection: Why Queues Model Fairness in Rwanda

Queues embody fairness because they follow the **First-In-First-Out (FIFO)** principle. In Rwanda's banking halls, transport hubs, and telecom centers:

- **Time-based fairness:** Whoever arrives first is served first.
- **Transparency:** Everyone sees their position and progress.
- **No favoritism:** Reduces bias and corruption.

- **Reduces conflict:** Clear order prevents disputes.

Queues reflect **Ubuntu** values—respect, patience, and community order