Overview: A hard disk drive consists of platters, tracks, sectors, and a read/write head. Disk I/O is performed sector by sector, and disk scheduling is crucial to optimizing performance.

Objectives of Disk Scheduling Algorithms:

1. **Maximize Throughput** – Increase the number of requests satisfied per unit time.
2. **Minimize Response Time** – Reduce the waiting time for each request.

Disk Scheduling Algorithms

1. FCFS (First Come, First Served)

- Serves requests in the order they arrive.
- No reordering of the queue.
- Ensures fairness (no starvation).
- **Disadvantage:** Poor performance due to long seek times.

2. SSTF (Shortest Seek Time First)

- Selects the request with the shortest seek time next.
- **Advantage:** Reduces total seek time compared to FCFS.
- **Disadvantage:** Starvation can occur if requests are concentrated in one area.

3. SCAN (Elevator Algorithm)

- Moves from the outermost track to the innermost while servicing requests, then reverses direction.
- **Advantage:** Reduces variance in response times.
- **Disadvantage:** May lead to longer waiting times for some requests.

4. LOOK (Modified SCAN)

- Works like SCAN but stops moving inward (or outward) when there are no more requests in that direction.
- **Advantage:** More efficient than SCAN as it avoids unnecessary movement.

5. C-SCAN (Circular SCAN)

- Moves inward servicing requests, then jumps to the outermost track and repeats.
- **Advantage:** Ensures fair wait times for all requests.
- **Disadvantage:** Can lead to higher seek times compared to LOOK.

6. C-LOOK (Circular LOOK)

- Moves in one direction servicing requests until there are none left, then jumps back to the outermost request.
- **Advantage:** Similar to C-SCAN but avoids unnecessary full sweeps.

Algorithm Steps:

1. Start.
2. Read the number of processes and requested tracks.
3. **FCFS:** Schedule processes in arrival order.
4. **SSTF:** Schedule the request closest to the current disk head position.
5. **SCAN:** Schedule from track 0 to the highest track, then reverse direction.
6. **C-SCAN:** Schedule from track 0 to the highest, then jump back to the lowest track.
7. Stop.

Summary:

- **FCFS:** Simple but inefficient.
- **SSTF:** Optimizes seek time but may cause starvation.
- **SCAN & LOOK:** Move in both directions; LOOK is more efficient.
- **C-SCAN & C-LOOK:** Circular versions ensuring fairness.

Choosing the best algorithm depends on workload characteristics and required performance balance.
