**OSL**

**ASSIGNMENT 8**

Name: Smeet Ramteke Class: TE 11

Roll No.: 33380 Batch: N11

**Aim**

Implement the C program for Disk Scheduling Algorithms: SSTF, SCAN, C-Look considering the initial head position moving away from the spindle.

**Theory**

Disk Scheduling

Disk scheduling is done by operating systems to schedule I/O requests arriving for the disk. Disk scheduling is also known as I/O scheduling. Disk scheduling is important because:

• Multiple I/O requests may arrive by different processes and only one I/O request can be served at a time by the disk controller. Thus other I/O requests need to wait in the waiting queue and need to be scheduled.

• Two or more request may be far from each other so can result in greater disk arm movement.

• Hard drives are one of the slowest parts of the computer system and thus need to be accessed in an efficient manner.

**Disk Scheduling Algorithms**

***First Come Forst Serve (FCFS) Algorithm***

• FCFS (First-Come-First-Serve) is the easiest disk scheduling algorithm among all the scheduling algorithms.

• In the FCFS disk scheduling algorithm, each input/output request is served in the order in which the requests arrive..

• In this algorithm, starvation does not occur because FCFS address each request.

*Advantages*

• In FCFS disk scheduling, there is no indefinite delay.

• There is no starvation in FCFS disk scheduling because each request gets a fair chance.

*Disadvantages*

• FCFS scheduling is not offered as the best service.

• In FCFS, scheduling disk time is not optimized.

***Shortest Seek Time First (SSTF) Algorithm***

• Basic idea is the tracks which are closer to current disk head position should be serviced first in order to minimise the seek operations.

*Advantages:*

• Better performance than FCFS scheduling algorithm.

• It provides better throughput.

• This algorithm is used in Batch Processing system where throughput is more important.

• It has less average response and waiting time.

*Disadvantages:*

• Starvation is possible for some requests as it favours easy to reach request and ignores the far away processes.

• Their is lack of predictability because of high variance of response time. • Switching direction slows things down.

***Elevator (SCAN) Algorithm***

• As the name suggests, this algorithm scans all the cylinders of the disk back and forth.

• Head starts from one end of the disk and move towards the other end servicing all the requests in between.

• After reaching the other end, head reverses its direction and move towards the starting end servicing all the requests in between.

• The same process repeats.

*Advantages:*

• It is simple, easy to understand and implement.

• It does not lead to starvation.

• It provides low variance in response time and waiting time.

*Disadvantages:*

• It causes long waiting time for the cylinders just visited by the head.

• It causes the head to move till the end of the disk even if there are no requests to be serviced.

***Circular LOOK (C-LOOK) Algorithm***

• Circular-LOOK Algorithm is an improved version of the LOOK Algorithm..

• Head starts from the first request at one end of the disk and moves towards the last request at the other end servicing all the requests in between.

• After reaching the last request at the other end, head reverses its direction.

• It then returns to the first request at the starting end without servicing any request in between.

• The same process repeats.

*Advantages:*

• It does not causes the head to move till the ends of the disk when there are no requests to be serviced.

• It reduces the waiting time for the cylinders just visited by the head. • It provides better performance as compared to LOOK Algorithm.

• It does not lead to starvation.

• It provides low variance in response time and waiting time.

*Disadvantages:*

• There is an overhead of finding the end requests.

**Conclusion :**

• In this assignment, we learnt about disk scheduling.

• We also learnt about the reason for requirement of disk scheduling. We also learnt about four disk scheduling algorithms along with there implementation and their comparison.