Eniselda Tusku - 21500228
Elena Cina - 21402992

For the purpose of this project we implemented a stimulation of four different disk scheduling algorithms in order to observe how they handle the scheduling for different sequence of requests and to calculate for each of them: ***the total time*** to serve all the requests , ***the average waiting time*** and ***the standard deviation of the waiting***.

In order to observe the efficiency and behaviour of each algorithm we generated three different sequences of requests and run several experiments. All experiments are run on Virtual Machine with 4 core

1. **First Sequence of Requests**

The first sequence we decided to show as part of this report is constructed in such a way that:

- No requests arrive at the same time.
- When each requests finishes, there is at least one other request that has already arrived and is waiting to be executed.

This sequence is designed in such a way with the intention to be a very simple sequence that does not involve any edge cases. Since we can control the number of requests in this case we decided to generate a sequence with **7** requests.

The sequence:

2 7
4 10
6 6
7 11
8 40
10 7
15 18

Size of sequence : 7

We run the sequence using each of the algorithms to see the differences between them and how they behave using the same input. Below is the output for all the used algorithms.

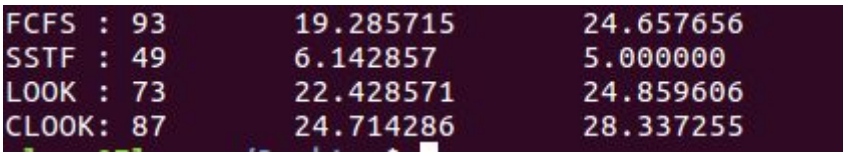


Figure 1. Output for first sequence

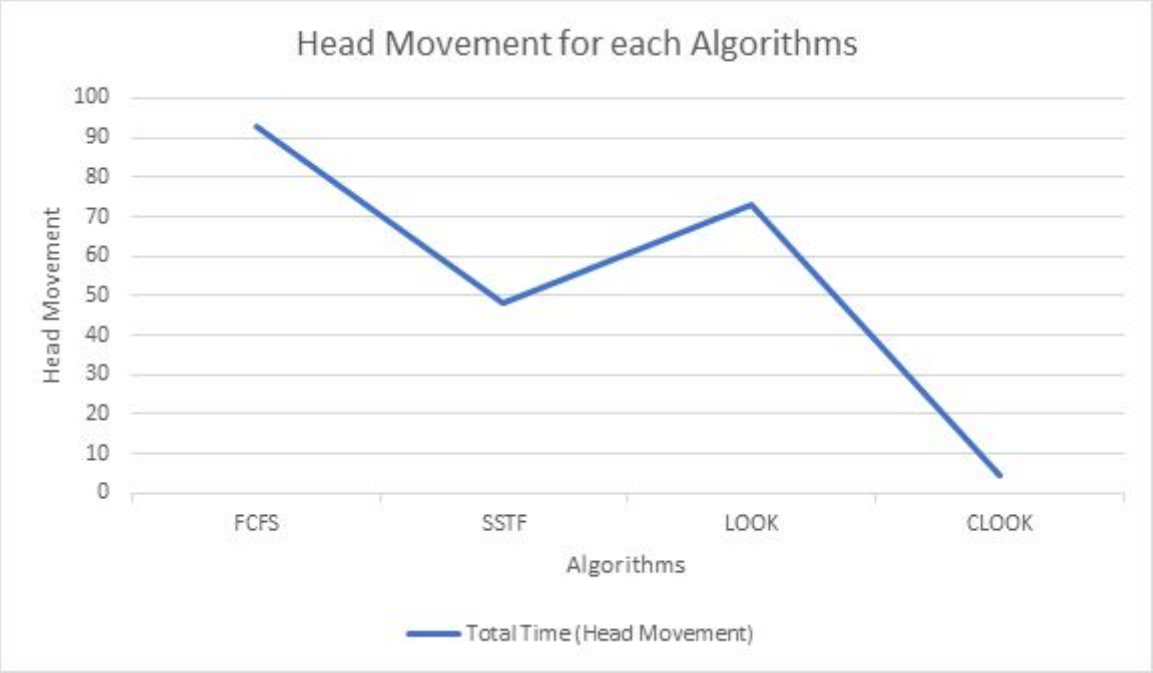
Analysis:

Using the principles of each disk scheduling algorithm the order in which they serve the cylinders of the sequence is as follows:

- FCFS: 7 , 10 , 6 , 11 , 40 , 7 , 18.
- SSTF: 7, 6, 10, 11, 7, 18, 40
- LOOK: 7, 10, 11, 40, 18, 7, 6
- CLOOK: 7, 10, 11, 40, 6, 7

As expected the order of the cylinders serving time is different for each algorithm. These differences have a great impact on the results of each of the measured variables. The following graphs demonstrate the differences that the algorithms have on ***the total time*** to serve all the requests , ***the average waiting time*** and ***the standard deviation of the waiting*** for the same sequence with 7 entries.

The graph regarding ***the total time*** *(head movement in terms of cylinders skipped over):*



The graph regarding ***the waiting time:***

The graph regarding ***the standard deviation :***

2. The second generated sequence of requests has a total of 10 requests. We generated this sequence in such a way that it covers all the edge cases we could come up with. These sequence has the following properties.

- There are 5 requests that arrive at the same time, at time 54.

- In the time that second request finishes being served, there is no other request in the queue.
- There is a demand for the same number of cylinder from two consecutive requests. At time 52 and 54 there are arriving requests for the same number of cylinder, which is cylinder 5.

The sequence:

5 10
7 12
50 30
51 7
52 5
54 5
54 20
54 6
54 21
54 15

Size of sequence : 10

Below we have provided the output for each algorithm.

Figure 2. Output for second sequence

Analysis:
Using the principles of each disk scheduling algorithm the order in which they will visit each cylinder is as follows:

- FCFS: 10, 12, 30, 7, 5, 20, 6, 21, 15
- SSTF: 10, 12, 30, 20, 15, 10, 7, 6, 5
- LOOK: 10, 12, 30, 20, 15, 10, 7, 6, 5
- CLOOK: 10, 12, 30, 5, 6, 7, 10, 15, 20

As it can be observed from the screenshot the output for each algorithm is different. The following graphs demonstrate the differences that the algorithms have on ***the total time*** to serve all the requests , ***the average waiting time*** and ***the standard deviation of the waiting*** for the same sequence with 10 entries.

The graph regarding ***the total time*** *(head movement in terms of cylinders skipped over):*

The graph regarding ***the waiting time:***

The graph regarding ***the standard deviation :***

Conclusion:

In this first part our aim was to get a better understanding on how different algorithms change with respect to the h**ead movement,** ***the average waiting time*** and ***the standard deviation*** on the same sequence and to observe if there was a stable correlation between them. As an be seen from the above results generally, the FCFS algorithm results in a higher number of head movements compared to others. Theoretically this is expected since this algorithm serves the cylinders only based on their arrival time which may result in great head movement when the cylinders are far from one another. For the other algorithms, as can be observed the change in the data is not stable and therefore no firm conclusion can be derived. Theoretically this is also expected since no algorithm is always more efficient than the others in terms of head movement, the average waiting time and the standard deviation. *Their efficency depends on the sequence provided.*

- In this part we have conducted experiments with different number of entries in our sequence and we concentrated on observing the behaviour of each algorithm individually to see how it changes when the sequence changes in size. We have used the data obtained from the 2 previously defined sequences and also the one defined below.

Third sequence of requests:

2 7
4 10
6 6

Size of sequence : 3

Output for third sequence.

Fourth sequence.

2 7
4 10
6 6
7 11
8 40
10 7
15 18
17 7
20 10
21 6
22 11
26 40
27 7
3018

Size of sequence : 14

Output for fourth sequence.

Analysis:

- ➔ The first Algorithm to be taken into consideration is FCFS. We have run this algorithms for all the sequences and have plotted the graph for each of the required data vs the number of requests. The results are as follows:

The graph regarding ***the total head movement:***

The graph regarding ***the mean of waiting time:***

The average of the head movement for FCFS is: 53

The average of the mean of waiting time for FCFS is: 28.675

The graph regarding ***the standard deviation:***

The average of standard deviation for FCFS is: 25.3775

➔ The second Algorithm to be taken into consideration is SSTF. We have run this algorithms for all the sequences and have plotted the graph for each of the required data vs the number of requests. The results are as follows:

The graph regarding ***the total head movement:***

The average of the head movements for SSTF is: 46.75

The graph regarding ***the mean of waiting time:***

The average of the mean of waiting time for SSTF is: 10.96

The graph regarding ***the standard deviation:***

The average of the standard deviation for SSTF is: 6.8

➔ The third Algorithm to be taken into consideration is LOOK. We have run this algorithms for all the sequences and have plotted the graph for each of the required data vs the number of requests. The results are as follows:

The graph regarding ***the total head movement:***

The average of the head movement for LOOK is: 53.25

The graph regarding ***the mean of waiting time:***

The average of the mean of waiting for LOOK is: 85.32

The graph regarding ***the standard deviation:***

The average of the standard deviation for LOOK is: 16.5

- ➔ The fourth Algorithm to be taken into consideration is CLOOK. We have run this algorithms for all the sequences and have plotted the graph for each of the required data vs the number of requests. The results are as follows:

The graph regarding ***the total head movement:***

The average of the head movement for CLOOK is: 63.5

The graph regarding ***the mean of waiting time:***

The average of the mean of waiting time for CLOOK is: 25.305

The graph regarding ***the standard deviation:***

The average of the standard deviation for CLOOK is: 25.76

Conclusion: *This conclusion applies to all the algorithms*.

As expected the different sequences generated different values for each of the parameters: head movement, mean of waiting time and standard deviation. As can be seen from the graphs the correlation between these parameters and the Number of requests is not a stable one. For example in the LOOK algorithm the number of head movements is smaller in the sequence with size 7 compared to the ones with size 10 and 14 where 14 is the smallest one. However if we see the mean waiting time this result is reversed and 14 has the larger waiting time. The algorithms like FCFS seem to be increasing in all three of these parameters when the number of request increases, but this result cannot lead to a fixed statement. The reason for this is because we may have a small number of requests but they correspond to cylinders which are far from one another and come at very different times (when one finishes the other in line has not arrived yet). In this case even though we have a small number of requests the number of head movement and waiting time will be large. On the other hand we may have a sequence with a large number of requests but whose elements are close to one another and arrive at similar times. In this case even though the number of requests is large the number of head movement and the waiting time will be small. The other algorithms also have the same behaviour.
Once again we conclude that the efficiency and the results of each algorithm with respect to h**ead movement, *the average waiting time*** and ***the standard deviation*** depends on the provided sequence and does not follow any fixed general rule.