Disk Scheduling Algorithms

Ref: https://www.geeksforgeeks.org/fcfs-disk-scheduling-algorithms/

Problem Statement

Write a program to implement following disk scheduling algorithms:

- First Come First Serve (FCFS)
- SCAN
- Circular – SCAN (C-SCAN)
- Shortest Seek Time First (SSTF).

Note: All the above algorithms will be implemented in a single menu- driven program.

Input Required

- Total number of cylinders (n)
- Initial head position (head)
- Total number of cylinder in request sequence (length of array of cylinder positions)
- Request sequence (array of cylinder positions)
- Direction of motion (optional for some algorithm)

Output

- Seek Sequence Calculations
- Seek Sequence
- Total Seek Time

Example: Sample Input

- Total cylinders = 200
- Initial head position = 50
- Total no. of cylinder in request sequence= 8
- Request sequence:

  176, 79, 34, 60, 92, 11, 41, 114

Example: Sample Output (FCFS)

- **Seek Sequence Calculations**

  Head movement: 176-50 = 126

  Head movement: 176-79 = 97

  Head movement: 79-34 = 45

  Head movement: 60-34 = 26

  Head movement: 92-60 = 32

  Head movement: 92-11 = 81

  Head movement: 41-11 = 30

  Head movement: 114-41 = 73

- **Seek Sequence:**

  176, 79, 34, 60, 92, 11, 41, 114

- **Total Seek Time** = 510 (126+97+45+26+32+81+30+73)

Algorithm for FCFS

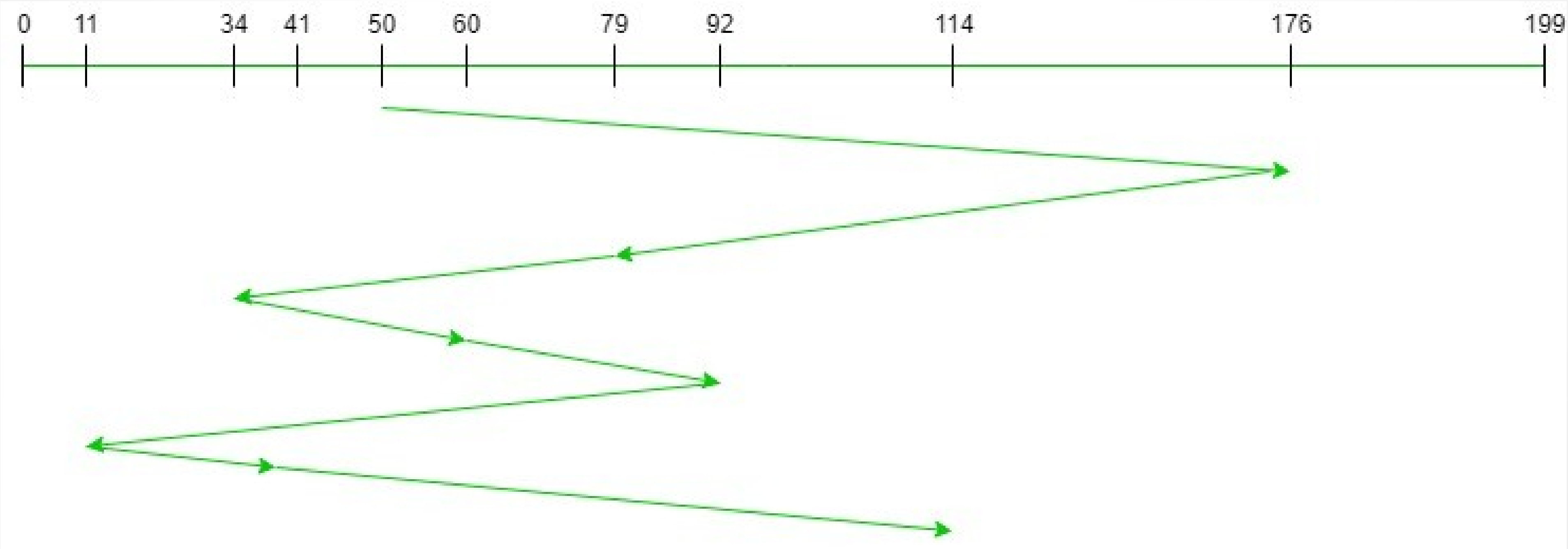
1. Let Request array represents an array storing indexes of cylinders that have been requested.
2. ‘head’ is the position of disk head.
3. One by one take the cylinders in default order and calculate the absolute distance of the cylinder from the head.
4. Increment the total seek time with this distance.
5. Currently serviced cyllinder position now becomes the new head position.
6. Go to step 3 until all cylinders in request array have not been serviced.

Example: FCFS

- Seek Sequence Calculations:

  (176-50) + (176-79) + (79-34) + (60-34) + (92-60) + (92-11) + (41-11) + (114-41) = 510

- Seek Sequence is 176, 79, 34, 60, 92, 11, 41, 114
- Total Seek Time = 510



Algorithm for SCAN

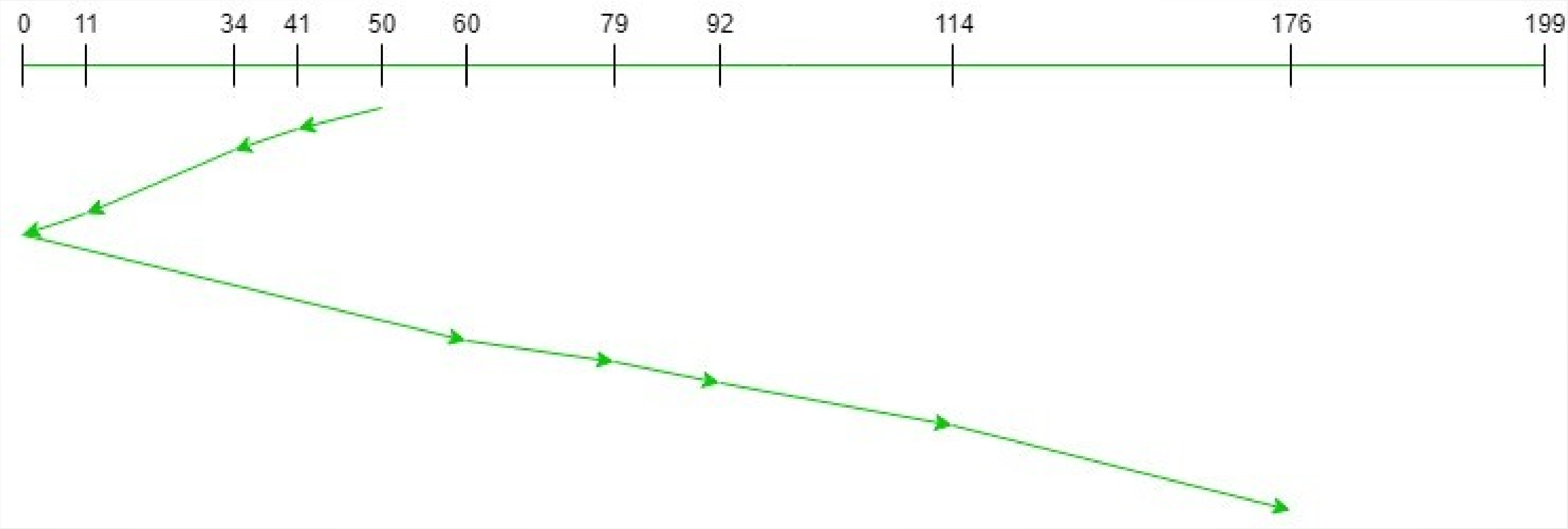
1. Let Request array represents an array storing indexes of cylinders that have been requested.
2. ‘head’ is the position of disk head.
3. Let direction represents whether the head is moving towards left or right.
4. In the direction in which head is moving service all tracks one by one.
5. Calculate the absolute distance of the cylinder from the head.
6. Increment the total seek time with this distance.
7. Currently serviced cylinder position now becomes the new head position.
8. Go to step 4 until we reach at one of the ends of the disk.
9. If we reach at the end of the disk reverse the direction and go to step 4 until all tracks in request array have not been serviced.

Example: SCAN

- Seek Sequence Calculations:

  (50-41) + (41-34) + (34-11) + (11-0) + (60-0) + (79-60) + (92-79) + (114-92) + (176-114) = 226

- Seek Sequence is 41, 34, 11, 60, 76, 92, 114, 176
- Total Seek Time = 226



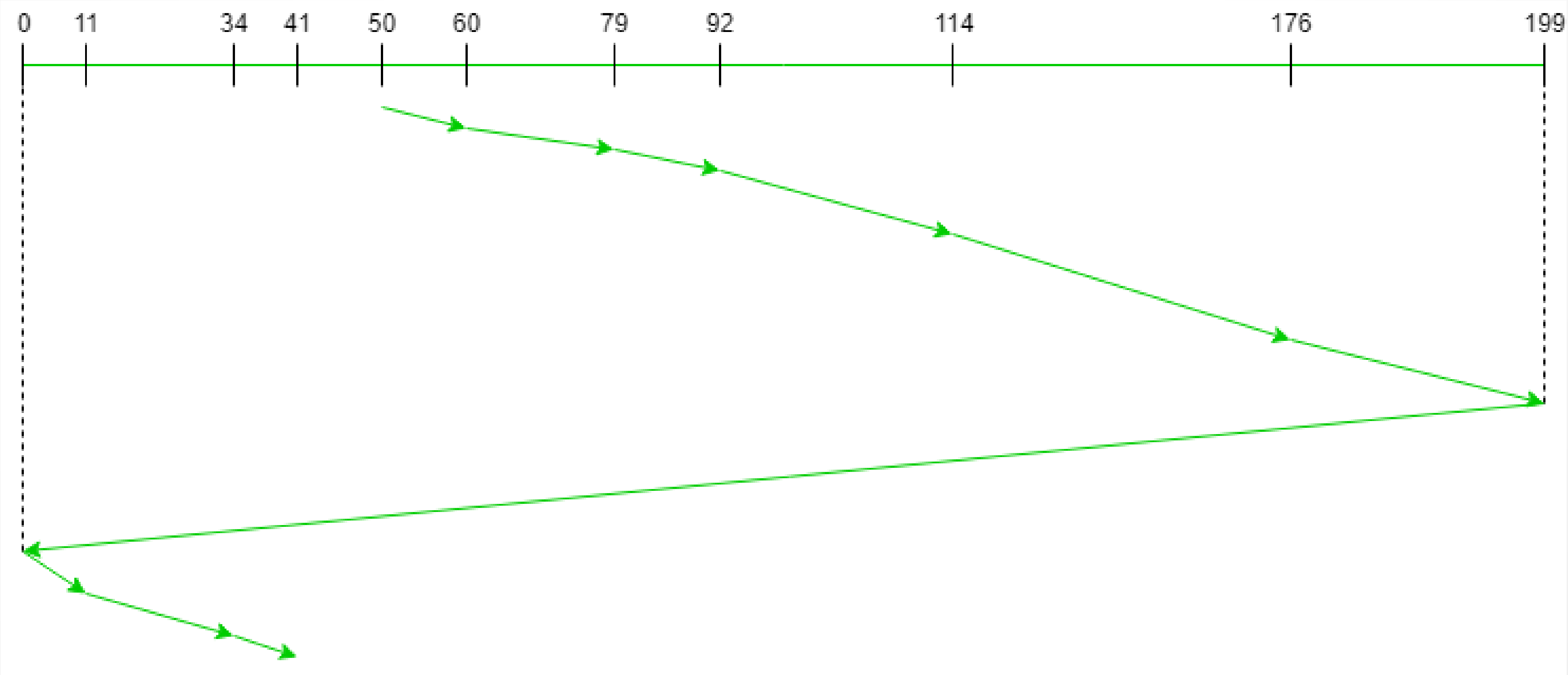
Algorithm for C-SCAN

1. Let Request array represents an array storing indexes of cylinders that have been requested.
2. ‘head’ is the position of disk head.
3. Let direction represents whether the head is moving towards left or right.
4. In the direction in which head is moving service all cylinders one by one.
5. Lets assume, the head services only in the right direction from 0 to size of the disk.
6. While moving in the left direction do not service any of the cylinders requests.
7. When we reach at the beginning (left end) reverse the direction.
8. While moving in right direction it services all cylinders requests one by one.
9. While moving in right direction calculate the absolute distance of the cylinder from the head.
10. Increment the total seek time with this distance.
11. Currently serviced cylinder position now becomes the new head position.
12. Go to step 9 until we reach at right end of the disk.
13. If we reach at the right end of the disk reverse the direction and go to step 6 until all cylinders in request array have not been serviced.

Example: C-SCAN

- Seek Sequence Calculations:

  (60-50) + (79-60) + (92-79) + (114-92) + (176-114) + (199-176) + (199-0) + (11-0) + (34-11) + (41-34) = 190
- Seek Sequence is 60, 79, 92, 114, 176, 11, 34, 41
- Total Seek Time = 190



Algorithm for SSTF

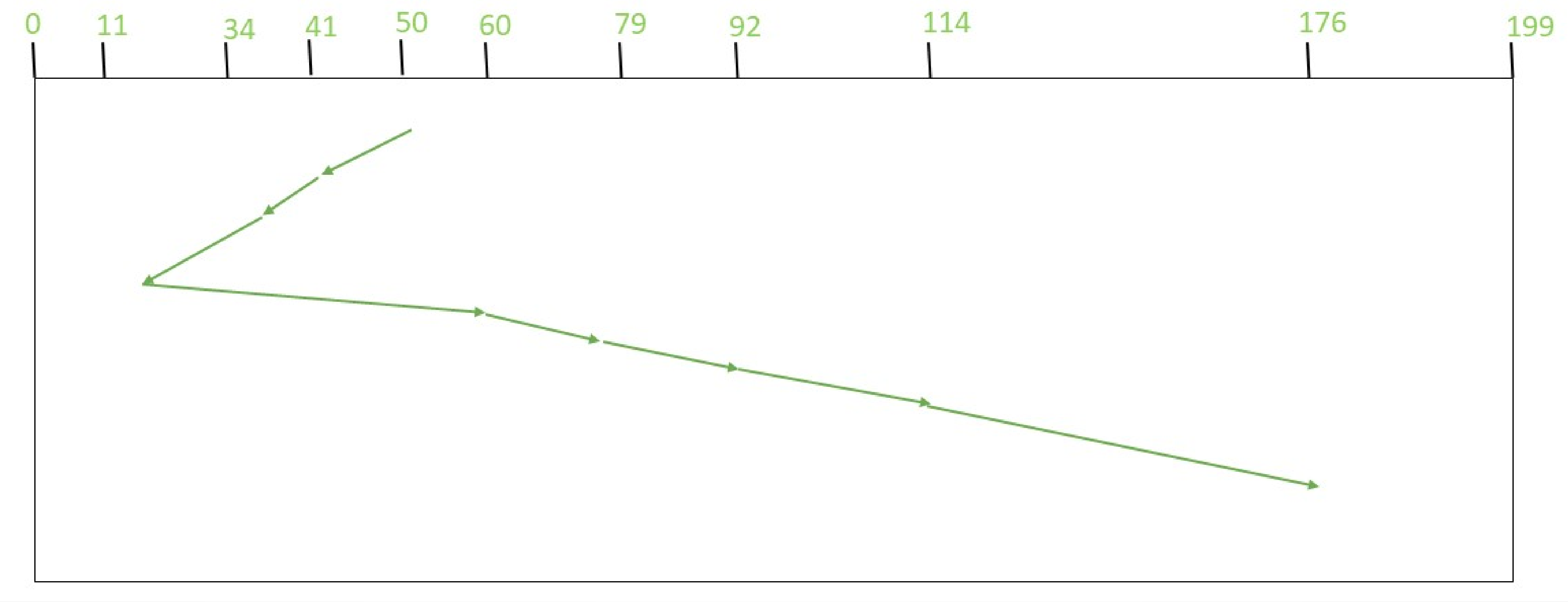
1. Let Request array represents an array storing indexes of cylinders that have been requested.
2. ‘head’ is the position of disk head.
3. Find the positive distance of all cylinders in the request array from head.
4. Find a cylinder from requested array which has not been accessed/serviced yet and has minimum distance from head.
5. Increment the total seek time with this distance.
6. Currently serviced cylinder position now becomes the new head position.
7. Go to step 3 until all cylinders in request array have not been serviced.

Example: SSTF

- Seek Sequence Calculations:

  (50-41) + (41-34) + (34-11) + (60-11) + (79-60) + (92-79) + (114-92) + (176-114) = 204

- Seek Sequence is 41, 34, 11, 60, 79, 92, 114, 176
- Total Seek Time = 204



Thank You!!