Parul® University
Vadodara, Gujarat

NAAC GRADE A++

Information and Communication Technology

I/O SYSTEMS, FILE & DISK MANAGEMENT:

Study Guide

Assistant Prof. Sumersing Patil
CSE-AI&DS, PIET
Parul University

Disk Arm Scheduling Algorithm

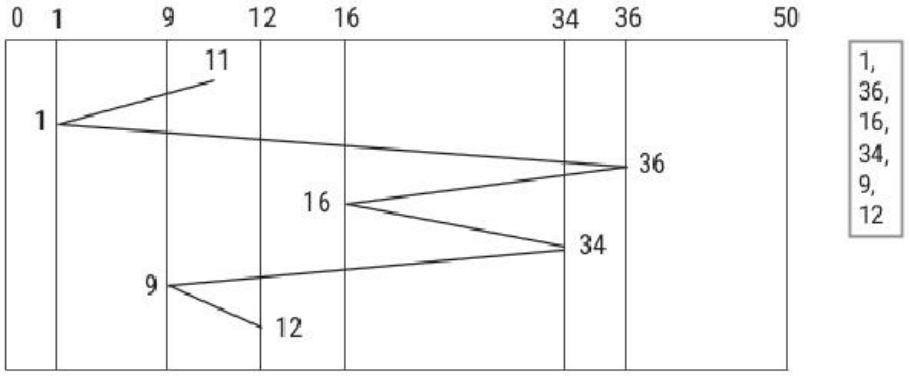
- Following are different types of Disk Arm Scheduling Algorithm
  1. FCSC (First Come First Serve) / FIFO (First In First Out)
  2. SSTF (Shorted Seek Time First)
  3. SCAN
  4. C-SCAN
  5. LOOK (Elevator)
  6. C-LOOK

Example for Disk Arm Scheduling Algorithm

- Consider an imaginary disk with 51 cylinders. A request comes in to read a block on cylinder 11. While the seek to cylinder 11 is in progress, new requests come in for cylinders 1, 36, 16, 34, 9, and 12, in that order.
- Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk scheduling Algorithms?

FCSC (First Come First Serve) / FIFO (First In First Out)

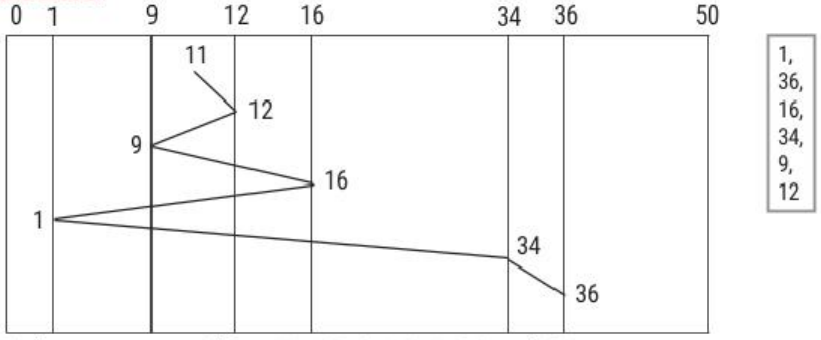
- Here **requests are served in the order of their arrival**.



- Disk movement will be 11, 1, 36, 16, 34, 9 and 12.
- Total cylinder movement: (11-1) + (36-1) + (36-16) + (34-16) + (34-9) + (12-9) = **111**

SSTF (Shortest seek time first)

- We can minimize the disk movement by serving the **request closest to the current position of the head.**



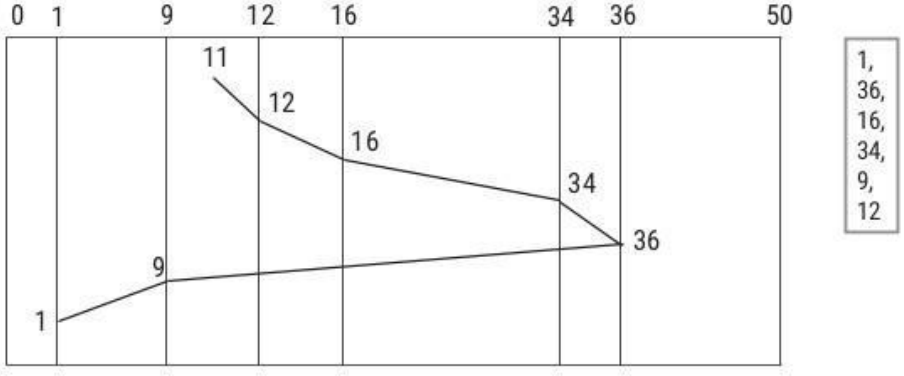
- Disk movement will be 11, 12, 9, 16, 1, 34 and 36.
- Total cylinder movement: (12-11) + (12-9) + (16-9) + (16-1) + (34-1) + (36-34) = 61

LOOK

- Keep **moving in the same direction until there are no more outstanding requests pending** in that direction, **then algorithm switches the direction.**
- **After switching the direction the arm will move to handle any request on the way.** Here first it moves in up direction then goes in down direction.
- In this algorithm, the **software maintains 1 bit**: the current direction bit, which takes the value either UP or DOWN.

LOOK

- Here first it moves in up direction then goes in down direction



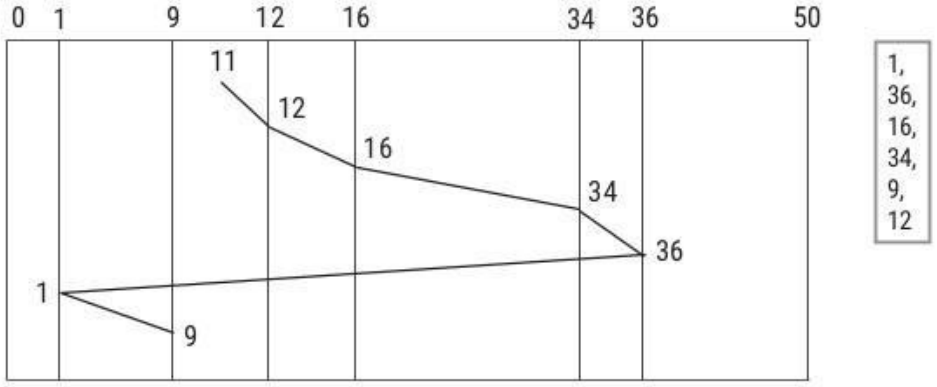
- Disk movement will be 11, 12, 16, 34, 36, 9 and 1.
- Total cylinder movement: (12-11) + (16-12) + (34-16) + (36-34) + (36-9) + (9-1) = 60

C-LOOK

- Keep moving in the same direction until there are no more outstanding requests pending in that direction, then algorithm switches direction.
- **When switching occurs the arm goes to the lowest numbered cylinder with pending requests and from there it continues moving in upward direction again.**

C-LOOK

- Here first it moves in up direction then goes in down direction



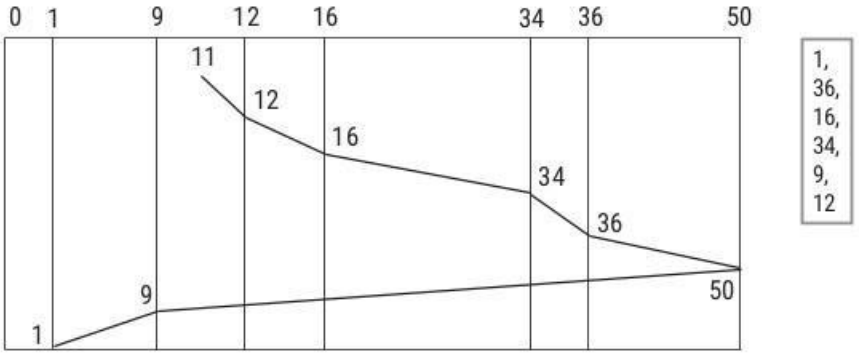
- Disk movement will be 11, 12, 16, 34, 36, 1 and 9.
- Total cylinder movement: (12-11) + (16-12) + (34-16) + (36-34) +(36-1)+(9-1) = 68

SCAN

- **From the current position disk arm starts in up direction and moves towards the end**, serving all the pending requests until end.
- At that end arm **direction is reversed (down) and moves towards the other end serving the pending requests on the way.**
- This is also called as **elevator algorithm.**

SCAN

- Here first it moves in up direction then goes in down direction



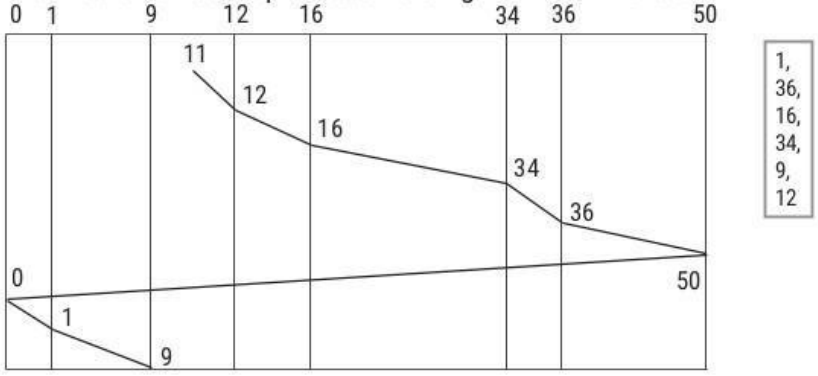
- Disk movement will be 11, 12, 16, 34, 36, 50, 9 and 1.
- Total cylinder movement: (12-11) + (16-12) + (34-16) +(36-34) +(50-36) + (50-9) + (9-1) = 88

C-SCAN

- **From the current position disk arm starts in up direction and moves towards the end**, serving request until end.
- At the end the arm **direction is reversed (down), and arm directly goes to other end and again continues moving in upward direction**.

C-SCAN

- Here first it moves in up direction then goes in down direction



- Disk movement will be 11, 12, 16, 34, 36, 50, 0, 1 and 9.
- Total cylinder movement: (12-11) + (16-12) + (34-16) +(36-34) +(50-36) + (50-0) + (1-0)+ (9-1) = 98

Example for Disk Arm Scheduling Algorithm [Exercise]

- Suppose that a disk drive has 200 cylinders, numbered 0 to 199 and order of request is: (82,170,43,140,24,16,190). And current position of Read/Write head is : 50.
- Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk scheduling Algorithms?
  - FCFS/FIFO (642)
  - SSTF (208)
  - SCAN (332)
  - C-SCAN (391)
  - LOOK (314)
  - C-LOOK (341)

File Management

- Operating system provides a platform to organize files in the form of directories so that we can easily navigate and manage them.
- O.S keeps track following information such as location, time, size, author etc.
- All the operations performed with respect to files are collectively called as File System or File Management.
- O.S helps us in performing various operations on the files such as,
  - Creating files.
  - Updating files.
  - Reading files.
  - Deleting files.
  - Protecting files from unauthorized access.

File

- A file is a unit of storing data on a secondary storage device such as a hard disk or other external media.
- Every file has a name and its data.
- Operating system associates various information with files. For example the date and time of the last modified file and the size of file etc....
- This information is called the file's attributes or metadata.
- The attributes varies considerably from system to system.

File attributes

- Below attributes tell who may access it and who may not:
  1. **Protection** - Who can access the file and in what way.
  2. **Password** - Password needed to access the file.
  3. **Creator** - ID of the person who created the file.
  4. **Owner** - Current owner.

File attributes

- Flags are bits or short fields that control or enable some specific property.
  1. **Read only flag** - 0 for read/write, 1 for read only.
  2. **Hidden flag** - 0 for normal, 1 for do not display the listings.
  3. **System flag** - 0 for normal, 1 for system file.
  4. **Archive flag** - 0 for has been backed up, 1 for needs to be backed up.
  5. **Random access flag** - 0 for sequential access only, 1 for random access.
  6. **Temporary flag** - 0 for normal, 1 for delete file on process exit.

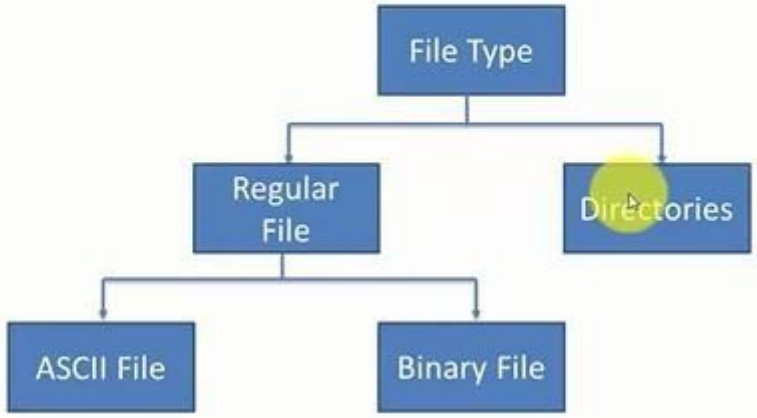
File operations

- Create
  - The purpose of the call is to announce that the file is coming and to set some attributes.
- Delete
  - When the file is no longer needed, it has to be deleted to free up disk space.
- Open
  - The purpose of the open call is to allow the system to fetch the attributes.
- Close
  - When all accesses are finished, the attributes and disk addresses are no longer needed, so the file should be closed to free up table space.

File operations

- Read
  - Data are read from file. Usually the bytes come from the current position.
- Write
  - Data are written to the file, usually at the current position.
- Append
  - This call is restricted form of write. It can only add data to the end of file.
- Seek
  - For a random access files, a method is needed to specify from where to take the data.
  - Seek repositions the file pointer to a specific place in the file.

Types of files



Regular file Vs Directories

- Regular File
  - Regular files are the ones that contain user information.
  - Example : Word file, excel file etc...
  - Application programs are responsible for understanding the structure and content of any specific regular file.
- Directories
  - Directories are system files for maintaining the structure of the file system.
  - To keep track of files, file systems normally have directories or folder.

Files access methods

- Sequential File Access
  - In Sequential access, process could read all the bytes or records from a file in order, starting at the beginning, but could not skip around and read them out of order.
  - Sequential files could be rewound, however, so they could be read as often as needed.
  - These files were convenient when the storage medium was magnetic tape or CD-ROM.

Files access methods

- Random File Access
  - Files whose bytes or records can be read in any order are called random access files.
  - Random access files are essentials for many applications, for example, data base systems.
  - Example: If an airline customer calls up and wants to reserve a seat on a particular flight, the reservation program must be able to access the record for that flight without having to read the records for thousands of other flights.

Directory Structure

- Directory structure are of:
  1. Single Level Directory System
  2. Hierarchical Directory System

Single Level Directory System

- The simplest form of directory system is having one directory containing all the files. This is some time called as root directory.
- Here directory contain three files.
- The advantages of this scheme are its simplicity and the ability to locate files quickly there is only one directory to look, after all.
- The single level is used for simple dedicated application, but for modern user with thousands of files, it would be impossible to find anything if all files were in a single directory.

Hierarchical Directory System

- In this system user can create an arbitrary number of subdirectories to organize their work.
- When the file system is organized as a directory tree, some way is needed for specifying file names.
- Two methods are used commonly
  1. Absolute path name
  2. Relative path name

Absolute path name

- An absolute path name consisting of the path from the root directory to the file.
- As an example, the path /usr /ast/mailbox means that the root directory contains a subdirectory usr, which in turn contains a subdirectory ast, which contain the file mailbox.

Relative path name

- Relative path name is used in conjunction with the concept of the working directory.
- User can designate one directory as the current working directory, in which case, all path names not beginning at the root directory are taken relative to working directory.
- For example if the current working is /usr/ast, then the file whose absolute path is /usr/ast/mailbox can be referenced simply as mailbox.
- Most operating systems that support a hierarchical directory system have two special entries in every directory,
  1. "." or "dot" refers to current directory
  2. ".." or "dotdot" refers to parent directory

Contiguous Allocation

- The simplest allocation scheme is to store each file as a contiguous run of disk block.