CM1604
Computer Systems Fundamentals

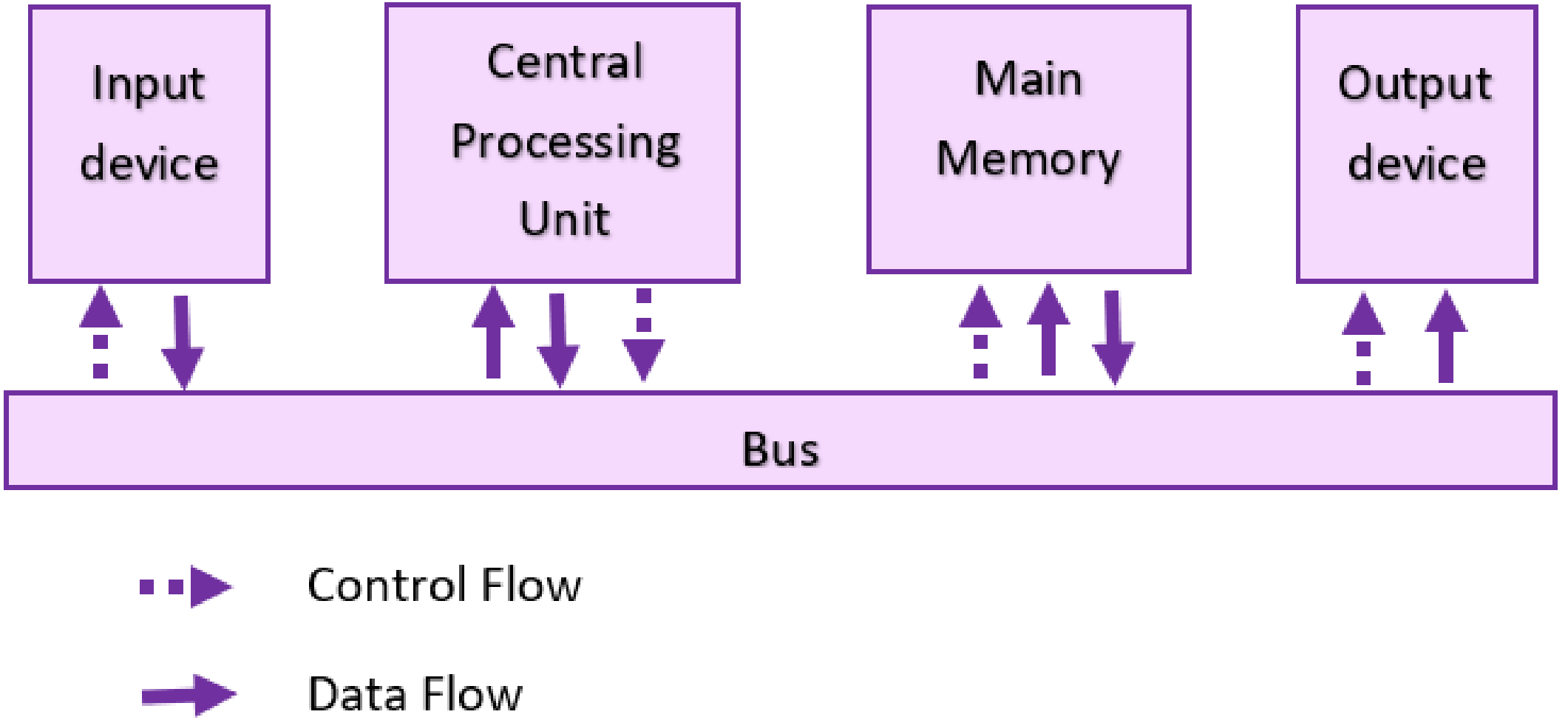
Process Management
File Management
Disk Scheduling

In this week lecture..

- Process Management
  - Process states
  - CPU Scheduling
- File Management
- Disk Scheduling

By the end of this lecture, you will:

- Understand different states of process
- Compare and contrast different CPU scheduling techniques
- Describe how the file management is done in OSs
- Compare and Contrast different disk scheduling algorithms


Input device
Central Processing Unit
Main Memory
Output device
Bus
Control Flow
Data Flow

Process Management & CPU Scheduling

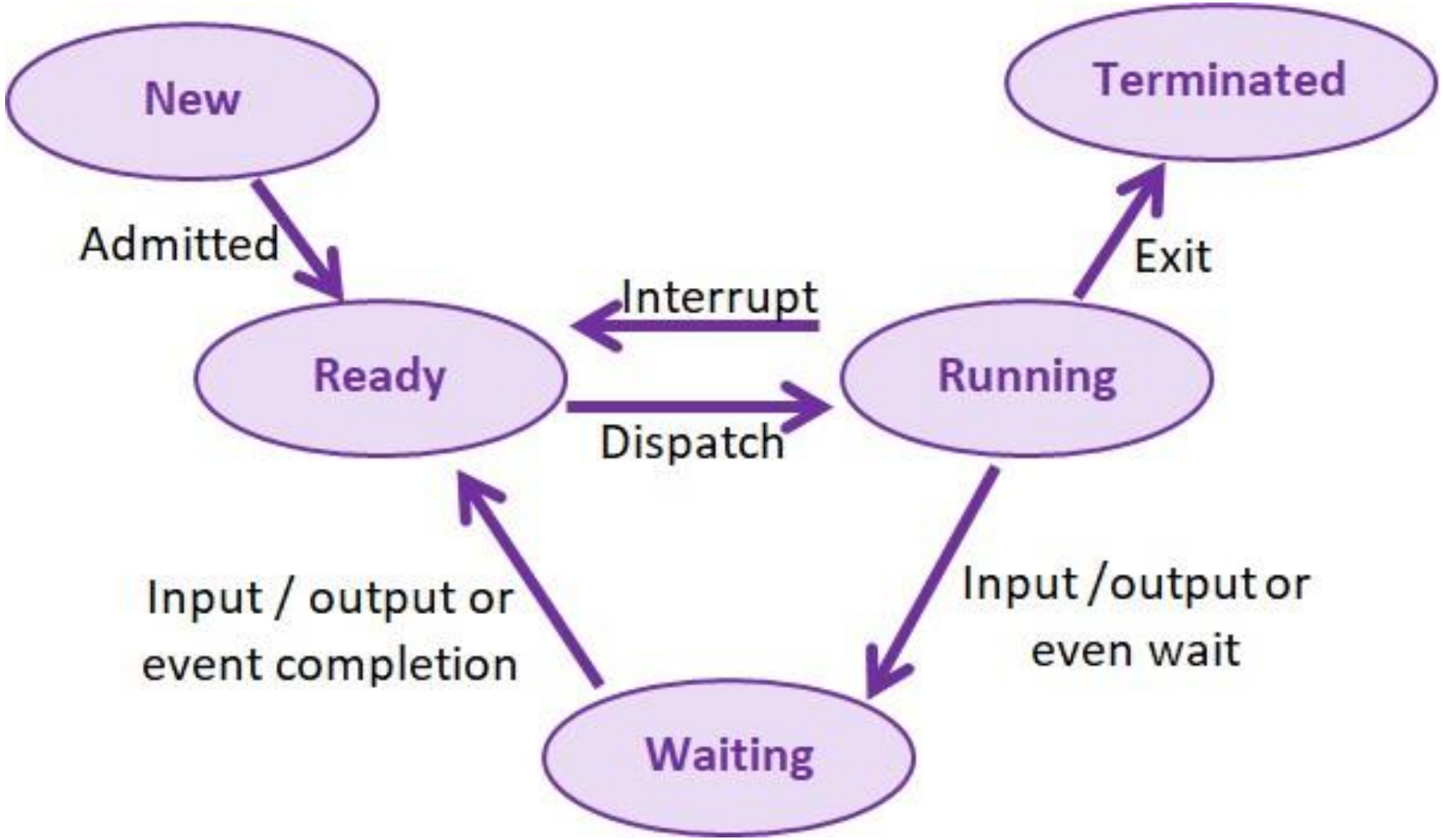
Process Management

- Process
  - Instance of a computer program in execution
  - Yet, single program can have multiple processes
- When a program is going to be executed
  - The program (machine code) gets loaded to the main memory from a storage device
  - Uses the CPU cycles to execute

Process Management …

- OS need to keep track of
  - What processes are running
  - What processes are in the memory
  - Method of scheduling the execution
- Therefore, each process store following information
  - Process Identification
  - Process State Information
  - Process Control Information
- This information is stored in a data structure called **Process Control Block (PCB)**

Process States



Process States

- **New**
  - A fresh new process that being created
  - No resources are allocated
- **Ready**
  - All the resources are allocated
  - Waiting for a chance to use the CPU
- **Running**
  - Currently being executed (fetch-execute cycle)
  - Only one process can be running in a processor

Process States …

- **Waiting**

  Waiting for resources other than CPU (input/output, memory page, a signal from another process)

- **Terminated**

  A completed process

  No need to keep track of resource allocation any more

CPU / Process Scheduling

- Process of determining which of the process in **ready** state to be moved to **running** state
  - There are multiple process in ready state
  - Only one process can be in running state

- Scheduling techniques may be ..
  - **Nonpreemptive** Scheduling
    The current executing process leaves the CPU voluntarily
  - **Preemptive** Scheduling
    The OS decides to put another process into ‘running’ state before the currently executing process finishes

CPU Scheduling ….

- **Turnaround Time (TAT)**

  Time interval from the time of submission of a process (first time the process enter the ready state) to the time of the completion of the process.
- **Burst Time (BT)**

  This is the time required by the process for its execution.
- **Waiting Time (WT)**

  The time spent by a process waiting in the ready queue for getting the CPU.

TAT = BT + WT

CPU Scheduling Algorithms

- **First-Come, First-Served**

  Processes are executed in the order in which they arrive into the ready state

- **Shortest Job Next**

  Process with shortest estimated running time in the ready state is executed first

- **Round Robin**

  Each process runs for a specified time slice and moves from the running state to the ready state to await till its next turn if not finished

CPU Scheduling

Process	Service time
P1	80
P2	160
P3	100
P4	30
P5	40
P6	110

First Come First Served

Process	Service time
P1	80
P2	160
P3	100
P4	30
P5	40
P6	110

FCFS	
Process	Service time
P1	80
P2	160
P3	100
P4	30
P5	40
P6	110

Shortest Job Next

Process	Service time
P1	80
P2	160
P3	100
P4	30
P5	40
P6	110

SJN	
Process	Service time
P4	30
P5	40
P1	80
P3	100
P6	110
P2	160

Round Robin

FCFS	
Process	Service time
P1	80
P2	160
P3	100
P4	30
P5	40
P6	110

File Management

File Systems

- **Main memory**

  Active programs and data are stored while it is used

  But volatile - lost when the power is turned off

- **Secondary memory**

  Non-volatile

  Used as the permanent storage to store data

- Commonly used secondary storage - magnetic disk drive

File Systems…

- **File**

  Named collection of related data that is used to organize secondary memory

  The smallest amount of data can be stored in the secondary memory -user view

- **File System**

  The way how the secondary memory is organized

  OS's logical view of the files that it manages

- **Directory**

  Named collection of files organized in a logical manner

File Classification

- **Text file**

  File contain only characters (ASCII or Unicode character set)

  Formatted as chunks of 8/16 bits and interpreted as characters

- **Binary file**

  File that contains data in specific format.

  Requires special interpretation - needs specific application

- Yet, all the information is sorted as binary in a computer

File Types

- **File Type**

  Specific type of information stored in a file

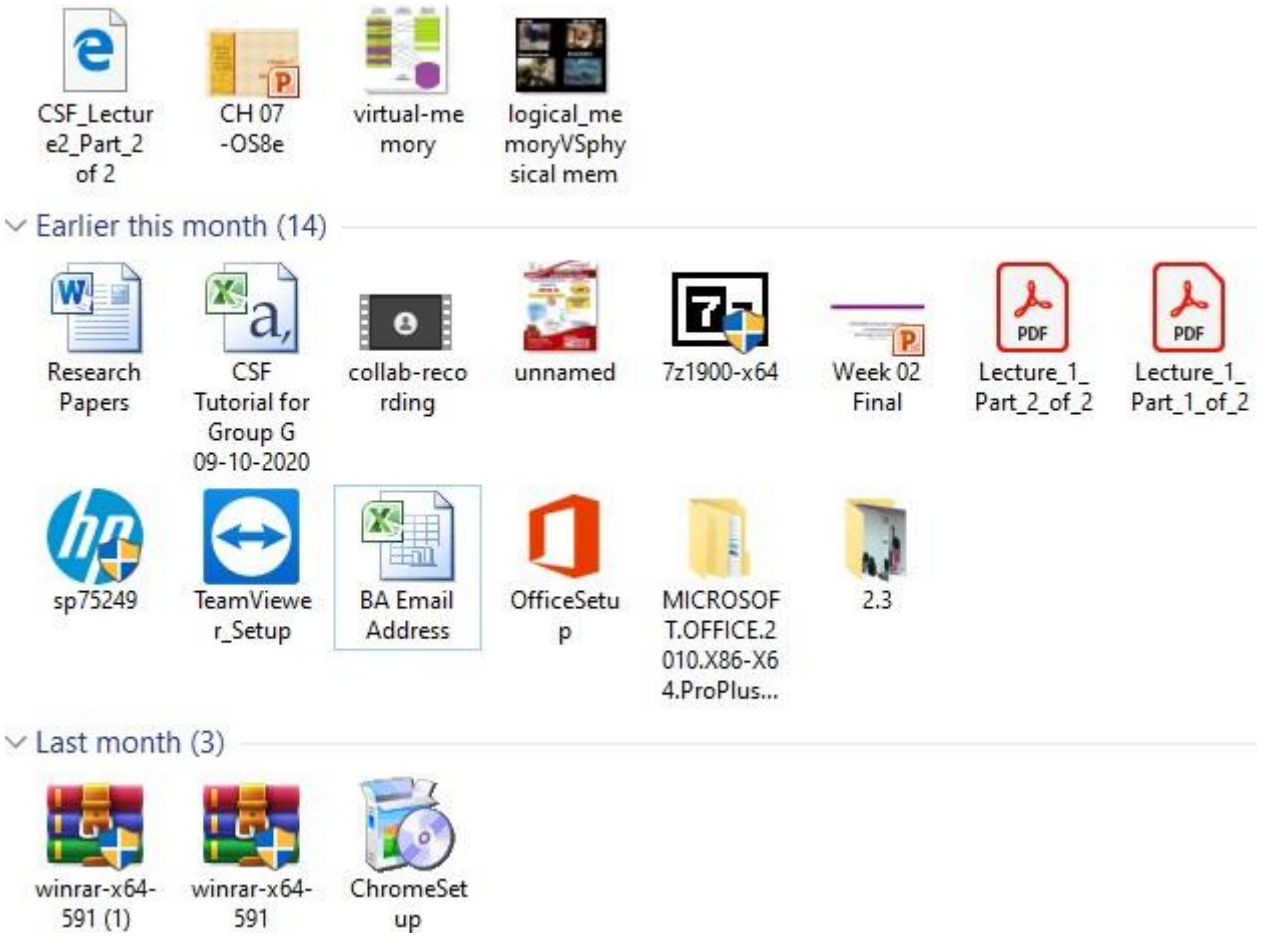
  Eg: JPEG, PDF, MP3

- **File Extension**

  Part of file name that indicates the file type

  Separated by a ‘.’ from the file name

File Types




```
BA Email Address.xlsx
CH 07 -OS8e.pptx
ChromeSetup.exe
collab-recording.mp4
CSF Tutorial for Group G 09-10-2020.csv
CSF_Lecture2_Part_2 of 2.html
Lecture_1_Part_1_of_2.pdf
Lecture_1_Part_2_of_2.pdf
logical_memoryVSphysical mem.jpg
MICROSOFT.OFFICE.2010.X86-X64.ProPlus.PurEvil
New MSc BDA students -04-10-2020 - list2 (1).xlsx
New MSc BDA students -04-10-2020 - list2.xlsx
OfficeSetup.exe
Research Papers.docx
sp75249.exe
TeamViewer_Setup.exe
unnamed.jpg
virtual-memory.png
Week 02 Final.pptx
winrar-x64-591 (1).exe
winrar-x64-591.exe
```

File Operations

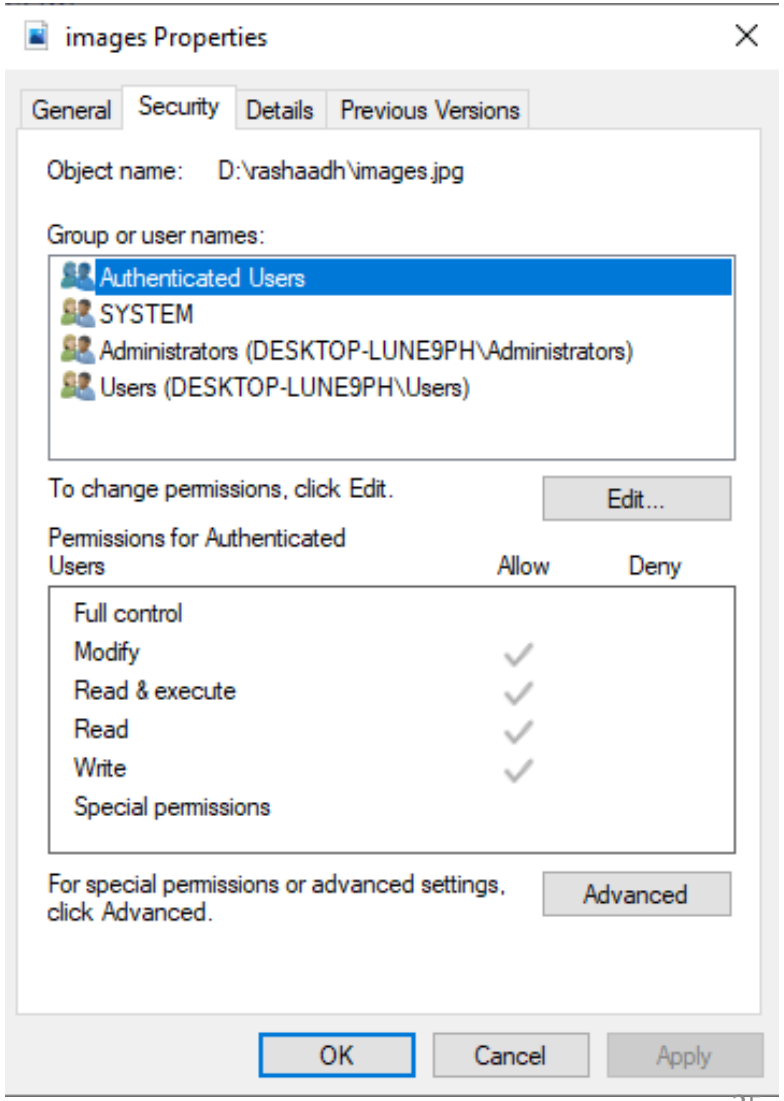
- File operations performed on files:

  create, delete,open, close, read, write, append, truncate, relocate, rename, copy

- OS provide mechanisms to perform the operations

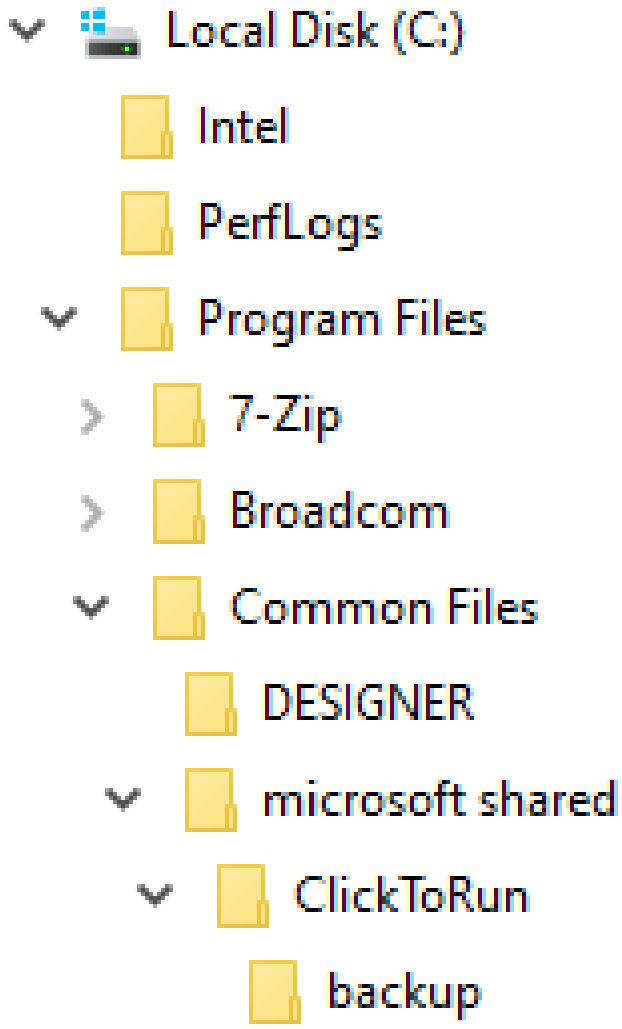
File Protection

- In multi-user system, multiple users store the files in the disk
- OS must ensure only the authorized users are allowed to access/do the file operation on files.



Directories

- Used to organize the files on disk
- Represented as a file in OS
- Contains the meta information about the files stored
- Can be nested to have a hierarchical file structure - easy to organize
- To visualize, file system is viewed as **directory tree**
- Directory at the highest level - **root directory**
- Subdirectory currently working - **working directory**



Path names

- **Path**

  Location of a file in the file system indicated using text

- **Absolute path**

  Path that begins from the root directory

- **Relative path**

  Path that begins at the current working directory

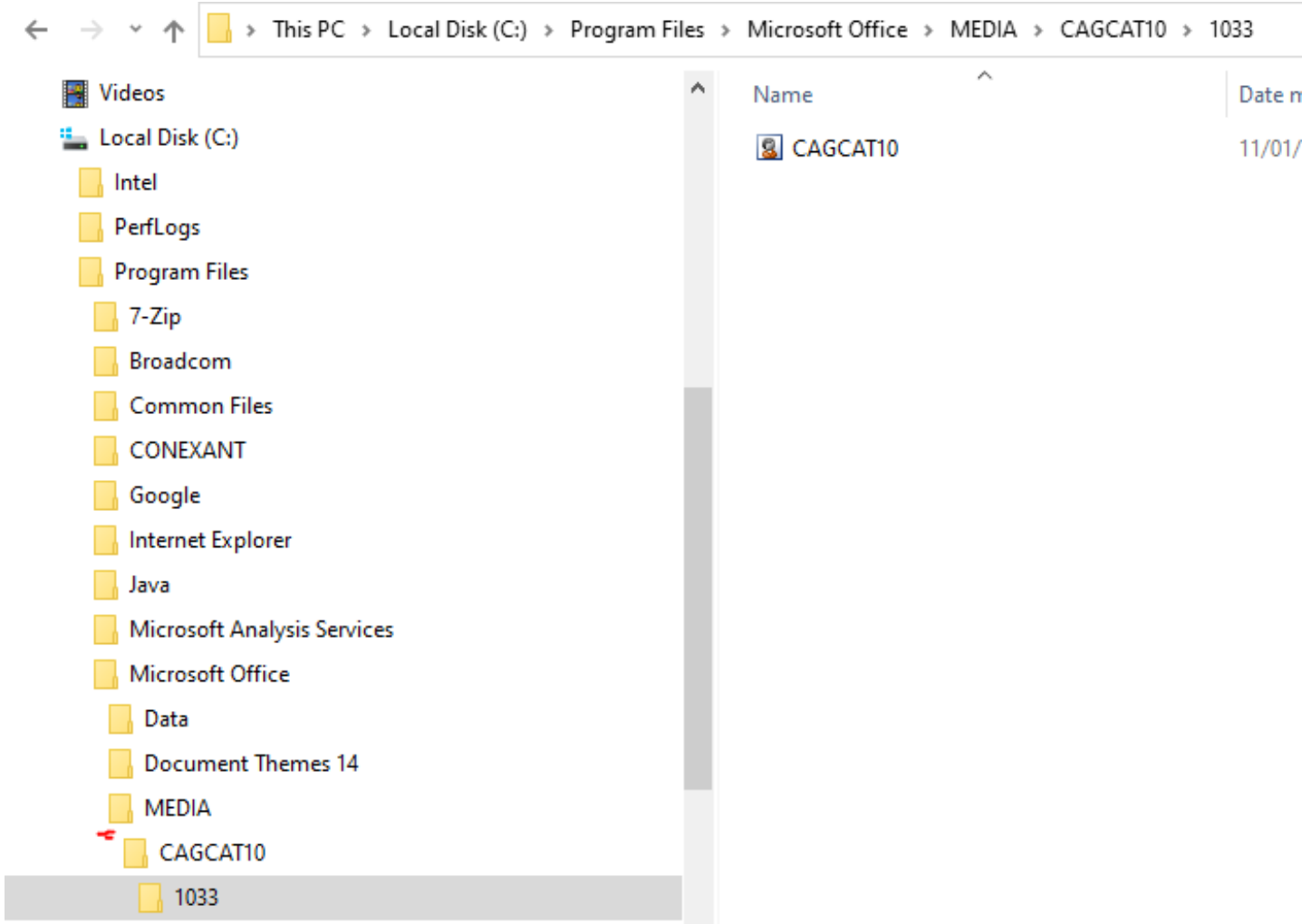
Path Names

Absolute path

C:\Program Files\Microsoft Office\MEDIA\CAGCAT10\1033

Relative path from “MEDIA”

CAGCAT10\1033



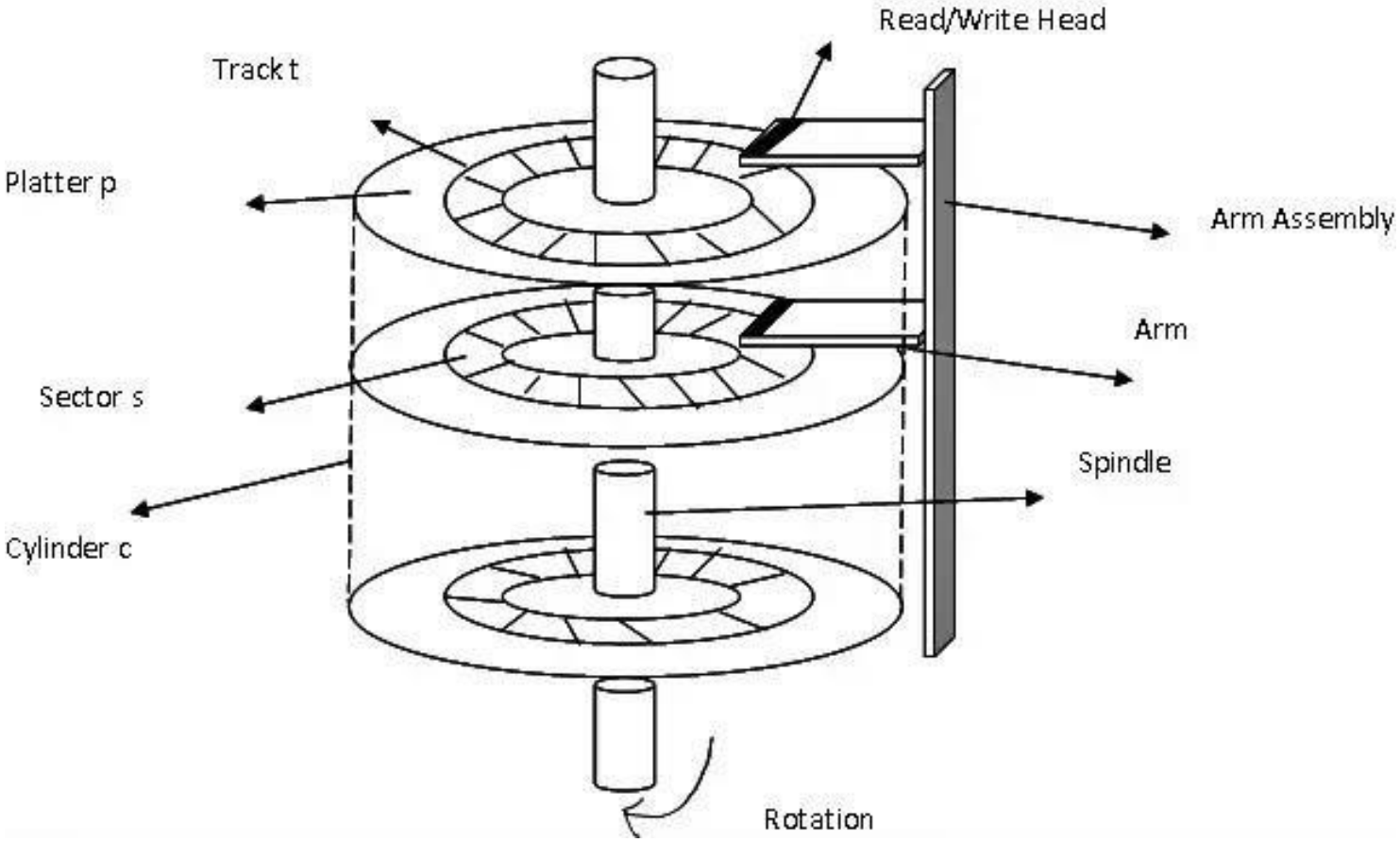
Disk Scheduling

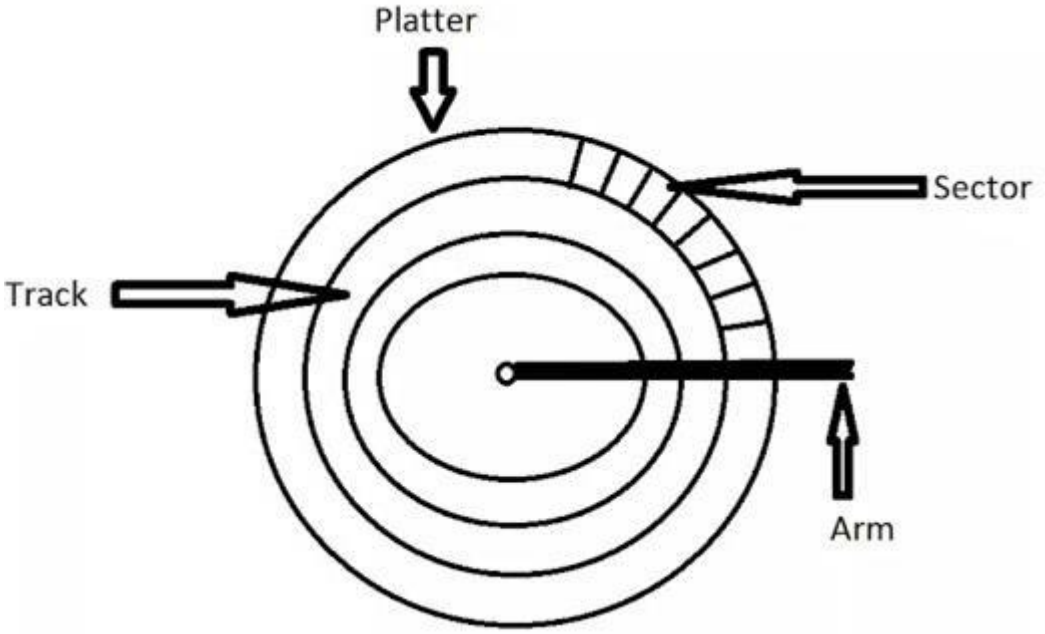
Disk scheduling

- Most important storage device in a computer is the magnetic disk
- For input/output the computer need to access the disk.
- This is the slowest operation in the computer.
- Therefore, need to decide on which request to cater in a way to make the task efficient.
- This process is called

Disk Scheduling / I/O scheduling

Disk Anatomy





https://geektech1717.blogspot.com/2020/05/disk-scheduling-operating-system.html

Disk Scheduling …

- Ways of moving the arm efficiently
- Goals:

  Fairness

    High throughout

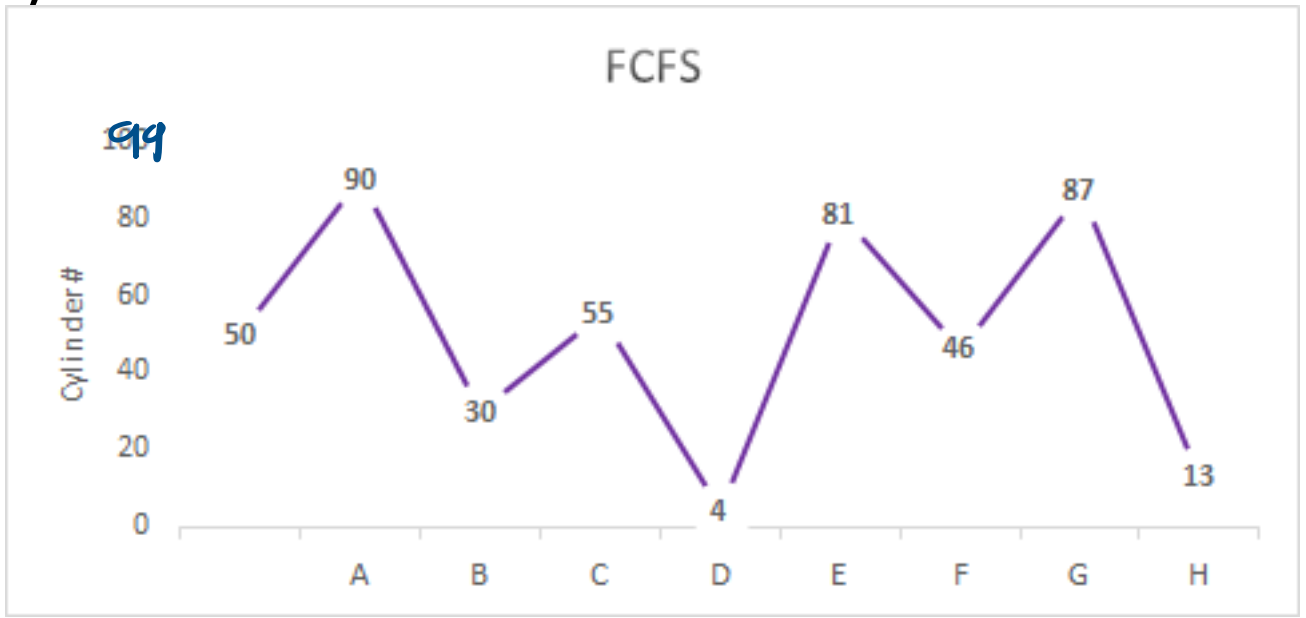
    Minimal travelling head time

  Eg: Disk has 100 tracks. Pointer is at 50.

    Order of request : 90, 30, 55, 4, 81, 46, 87, 13

First-Come First-Served (FCFS)

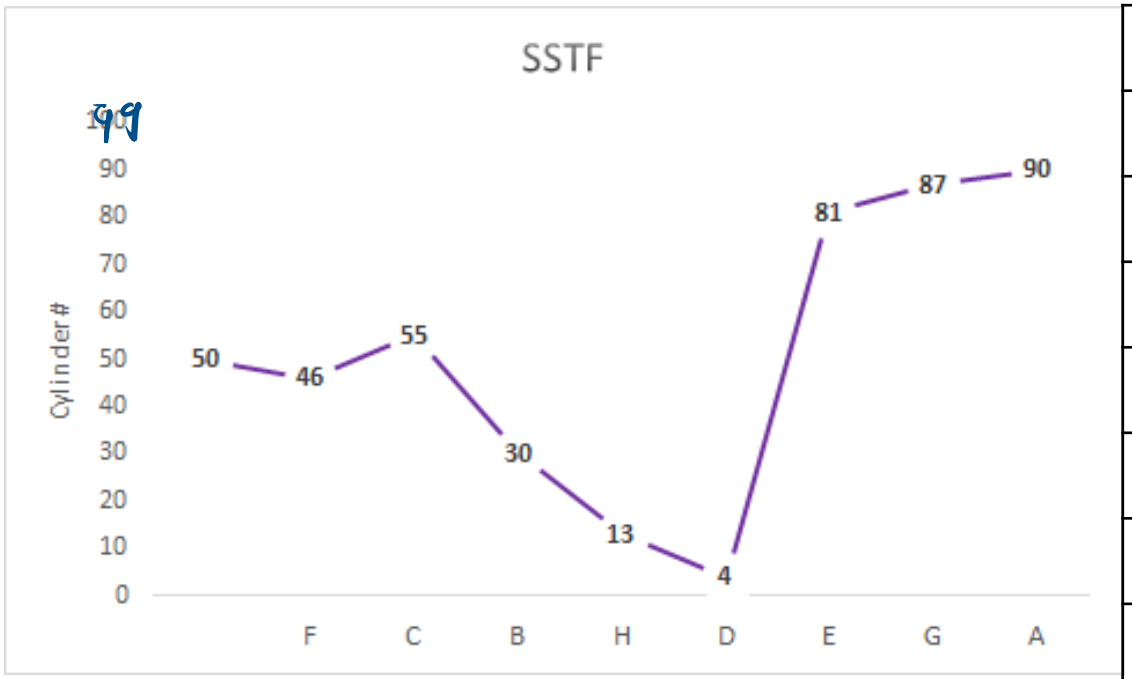
- Request served in the order they arrived
- Total seek time = (90-50) + (90-30) + (55-30) + (55-4) + (81-4) + (81-46) + (87-46) + (87-13)
- All get a fair opportunity
- No seek time optimization



A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

Shortest-Seek Time First (SSTF)

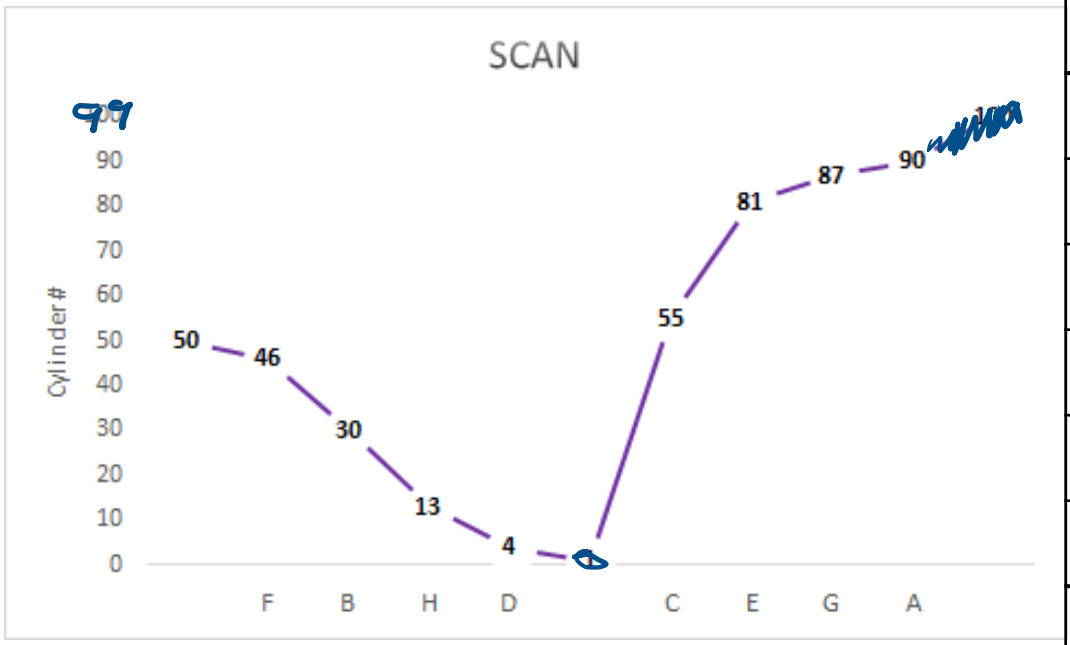
- Request with shortest seek time is executed each time
- Seek time = (50-46) + (55-46) + (55-30) + (30-13) + (13-4) + (81-4) + (87-81) + (90-87)
- Average response time decreases
- Overhead of calculation
- Starvation



A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

SCAN

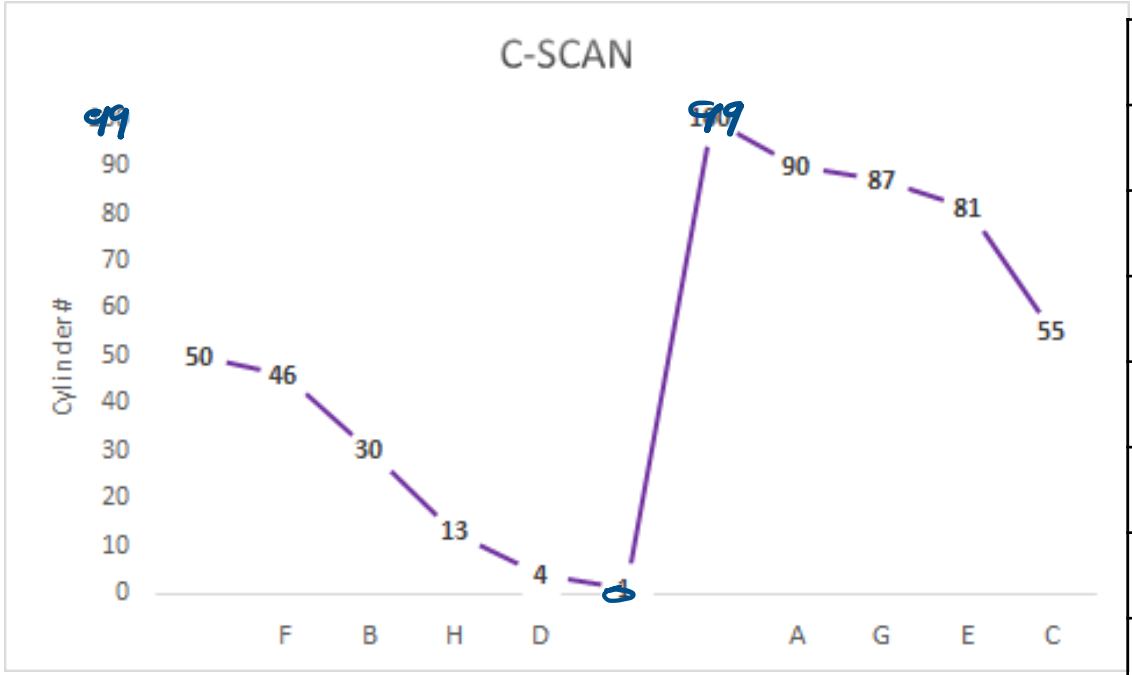
- Works as an elevator
- Arm moves in one direction upto the end and caters all the requests, then reverse the direction till the other end and service the requests
- Seek time = (50-0) + (90-0)
- requests at the middle are favoured



A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

C-SCAN

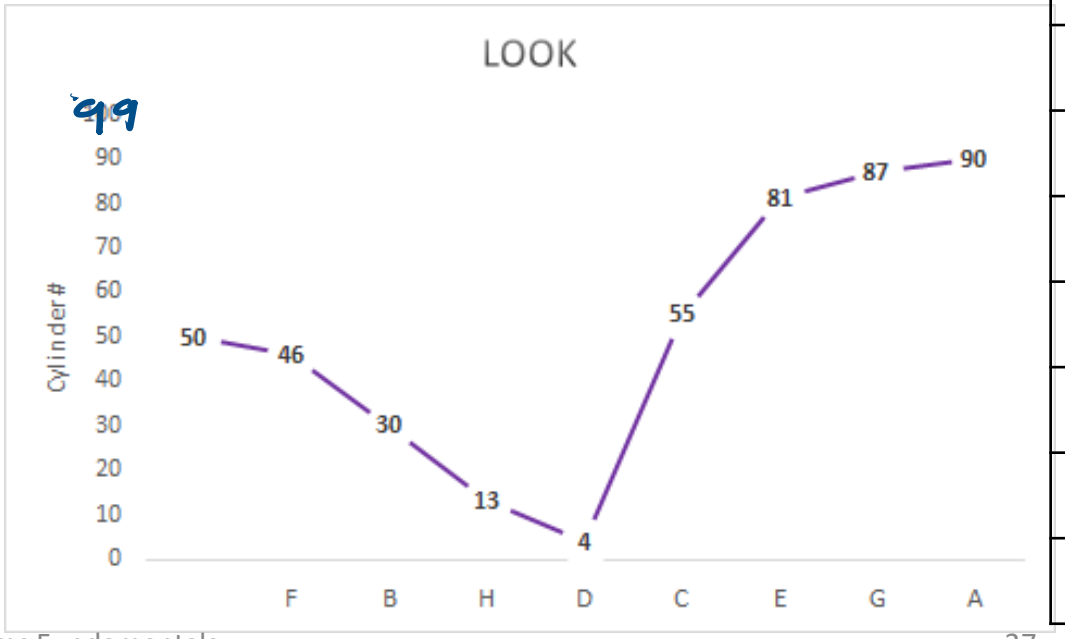
- Same as SCAN, but service the request only in one direction
- Seek time = (50-0) + (99-0) + (99-55)



A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

LOOK

- Same as SCAN, but does not reach the ends but till the last request
- Seek time = (50-4) + (90-4)
- Extra cost of reaching the edges is reduce



A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

C-LOOK

- LOOK version of C-SCAN.
- Seek time = (50-4) + (90-4) + (90-55)

A	90
B	30
C	55
D	4
E	81
F	46
G	87
H	13

REFERENCE

- Dale, N.B. and Lewis, J., 2007. Computer science illuminated. Jones & Bartlett Learning.
- Disk Scheduling - operating system tutorial https://geektech1717.blogspot.com/2020/05/disk-scheduling-operating-system.html

READING

Chapter # 10 and 11

- Computer science illuminated. Jones & Bartlett Learning.