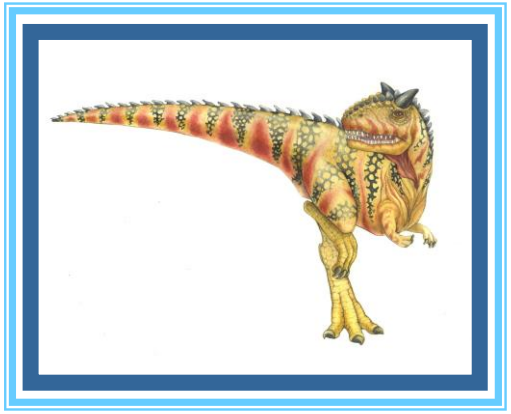
Xi'an Jiaotong-Liverpool University
西交利物浦大学

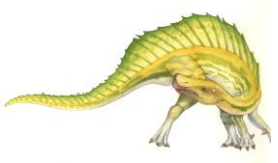
CPT104 - Operating Systems Concepts

STORAGE MANAGEMENT
Mass-Storage Systems



Mass-Storage Systems

- Overview
- Disk Organization/Management
- Disk Attachment
- Disk Scheduling
- Swap-Space Management
- RAID Structure



Overview

Secondary Storage

- Save data permanently.
- Slower than memory.
- Cheaper and greater than memory.

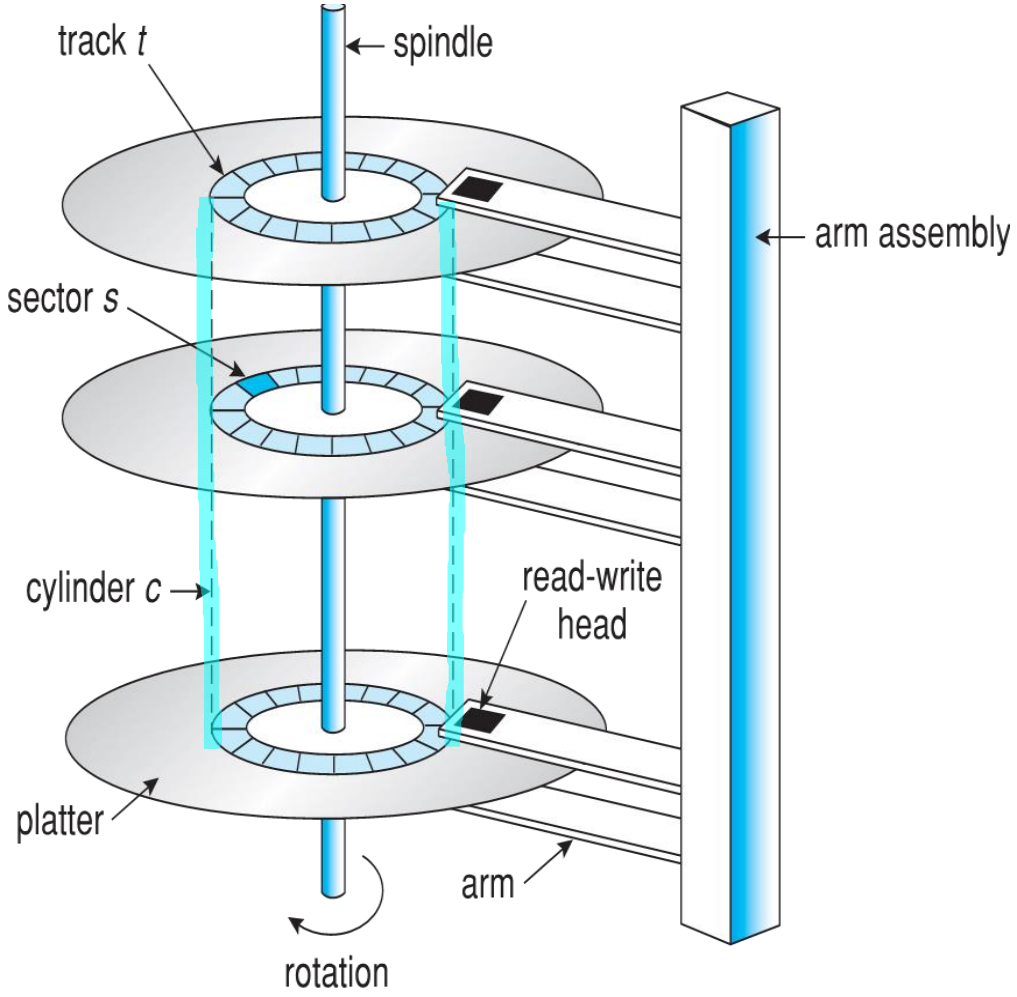
Types of Secondary Storages:

- **Sequential access devices** - store records sequentially, one after the other
  - data is stored in blocks or records
  - *Example*: magnetic-tape data storage, sequential access memory (SAM)
- **Direct access devices** - store data in discrete and separate location with a unique address.
  - *Example*: internal and external hard drives, magnetic disks, optical disks, flash memory, etc.

Disk Organization / Management

- Physical disk organization
- Logical disk organization

Physical Disk Organization



- a **platter** is a circular magnetic disk. Platters are typically made using an aluminium, glass or ceramic substrate.
- a **track** is a circular path on a platter surface.
- a group of tracks is called a **cylinder.**
- a track is logically divided into **sectors**. Each sector can hold a fixed amount of data, typically 512 bytes.
- **heads** are attached to a disk **arm** that moves all the heads as a unit.

Physical Disk Organization

Disk Management

When disk drives are originally manufactured, they contain no information.

The sectors that we want to access do not yet exist.

Low-level formatting, or **physical formatting** — **create sectors** on a blank platter

- each sector can hold **header** information, plus **data**, plus **error correction code** (**ECC**)

IMPORTANT!

A **sector** is a unit of data storage on a hard disk or other storage device (512 byte).

A **block** is a group of sectors on a hard disk or other storage device that are treated as a unit for data storage and retrieval purposes (1,024 bytes).

Logical Disk Organization

Disk Management

The logical structure of a disk refers to **the way information is organized and stored on the disk.**

- **Partitions -** are separate sections of the disk that act as independent units. Organize disk in one or more **groups of cylinders.**
- **Logical formatting** writes **file system data structures**
  - **File System –** method used to organize and store files on partitions.
  - **File Allocation Table** (FAT) - records the location of each file on the disk.
- **Boot sector / Master boot record (MBR)** the first part of the disk that is read when the computer is turned on and contains the code necessary to start the operating system (the bootstrap stored in read-only memory (ROM)).

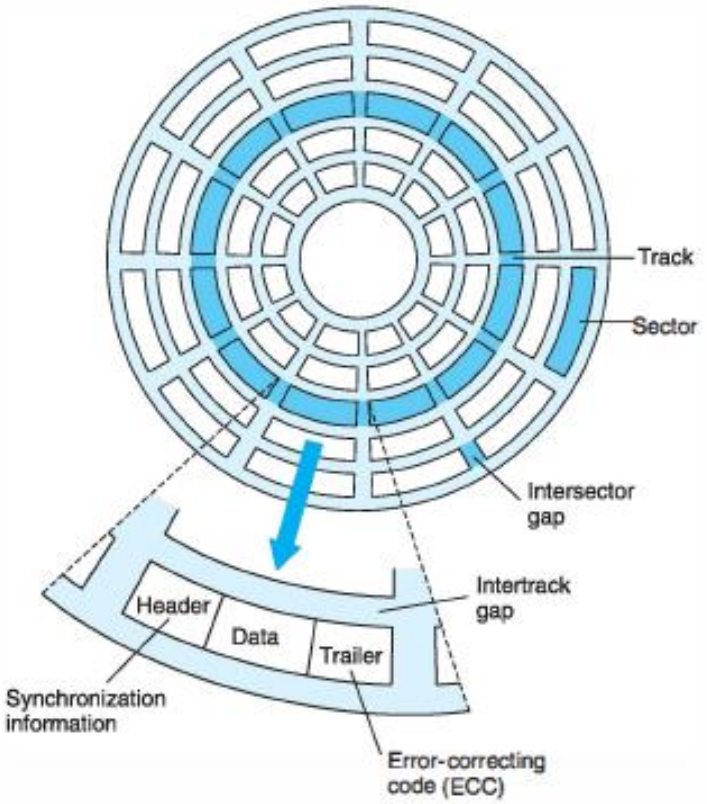
Disk Structure

- Disk is addressed as **one-dimension array of logical sectors.**
  - Logical sector **0**: the first sector of the first track of the first surface.
- ***Disk controller*** maps *logical sector* to *physical sector.*

The physical sector is what the hard drive actually reads and writes in;

The logical sector is what you can ask it to read or write in.

The physical sector/block size is always equal to or larger than the logical one.



Disk sector format

Disk speed

Transfer time = the time for data transfer between the drive and the computer.

Seek time = the time required to move the read/write head on the desired track/cylinder.

Rotational latency = the time taken to rotate the platter and bring the required disk sector under the read/write head.

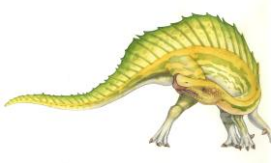
Positioning time / Random access time = seek time + rotational latency

Disk Access Time = the total time to perform any operation on the disk.

seek time + rotational latency + transfer time.

Mass-Storage Systems

- Overview
- Disk Organization/Management
- **Disk Attachment**
- Disk Scheduling
- Swap-Space Management
- RAID Structure



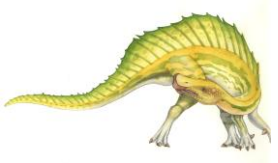
Disk Attachment

Computer systems can access disk storage:

- ***Host-attached storage*** - **via local I/O ports** (common on *small systems)*. The typical desktop PC uses:
  - *Integrated Drive Electronics* (IDE),
  - *Advanced Technology Attachment* (ATA),
  - *USB* (allows up to two drives per host controller).
- ***Network-Attached Storage*** - **via a remote-procedure-call (RPC) interface** (*distributed file systems).*
  - the RPCs are carried via TCP or UDP over an IP network.
- ***Storage Area Network****:* - reduces the latency of network communication (one drawback of Network-Attached Storage)
  - is a private network (**uses storage protocols**).
  - allows multiple hosts and multiple storage arrays to attach.

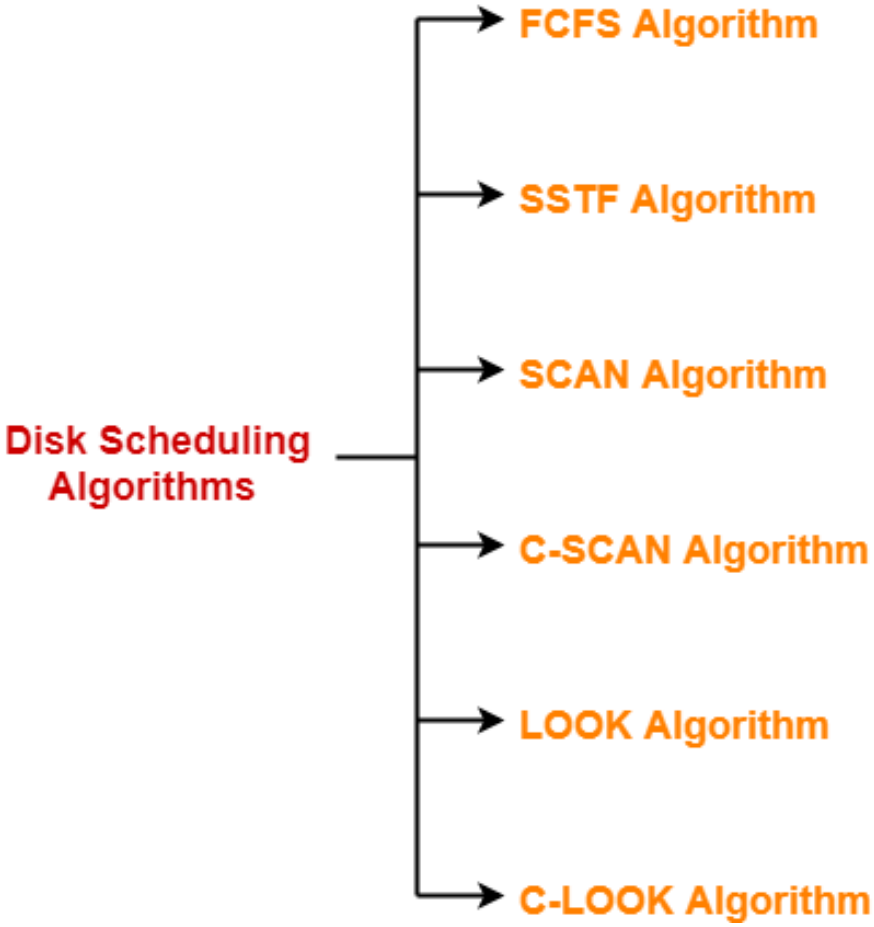
Mass-Storage Systems

- Overview
- Disk Organization/Management
- Disk Attachment
- **Disk Scheduling**
- Swap-Space Management
- RAID Structure
- Stable-Storage Implementation



Disk Scheduling

The disk scheduling is performed such that **the arm/head movement is utilized to service more I/O requests efficiently**.



Disk Scheduling

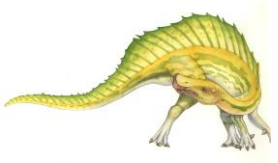
We illustrate scheduling algorithms with:

Consider a disk queue with I/O requests on the following cylinders in their arriving order:

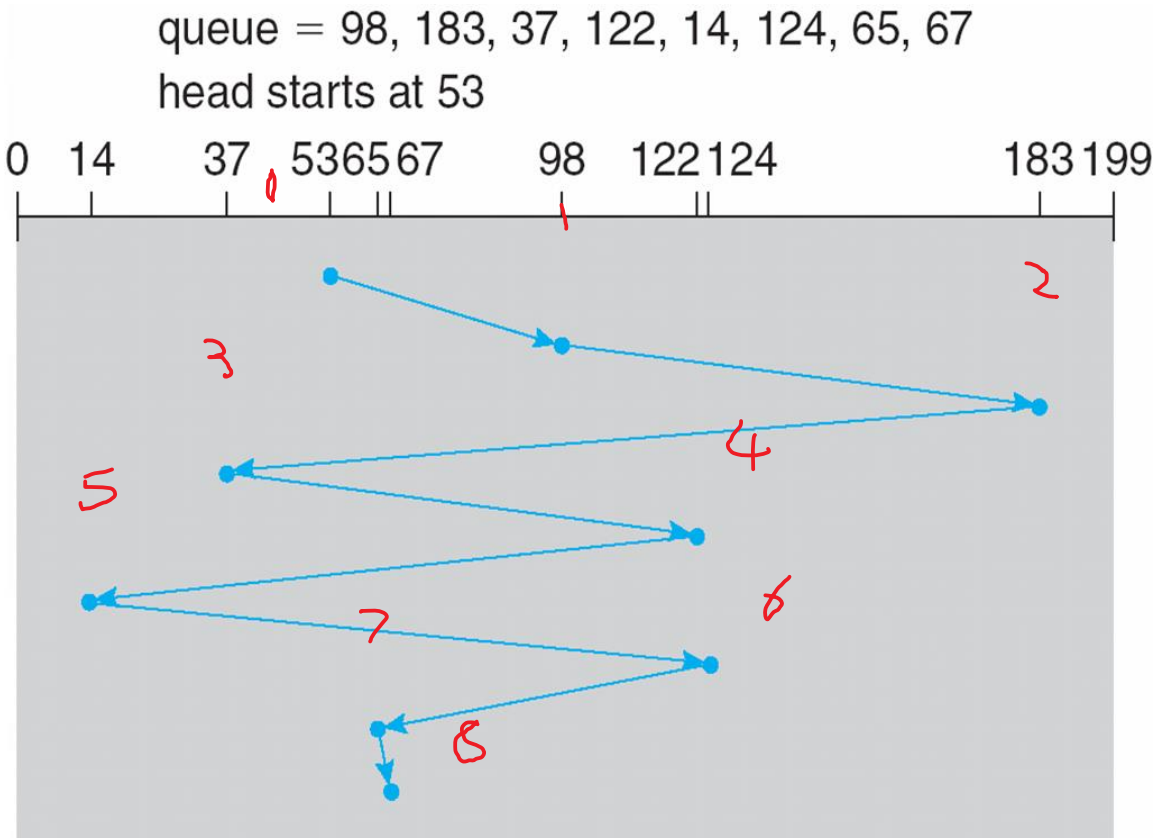
98, 183, 37, 122, 14, 124, 65, 67

The disk **head** is assumed to be at cylinder **53.**

The disk consists of total **200 cylinders** (0-199) .



First-Come First-Served FCFS



total head movement of 640 cylinders

Head movement:

= |53 – 98|

+ |98 – 183|

+ |183 – 37|

+ |37 – 122|

+ |122 – 14|

+ |14 – 124|

+ |124 – 65|

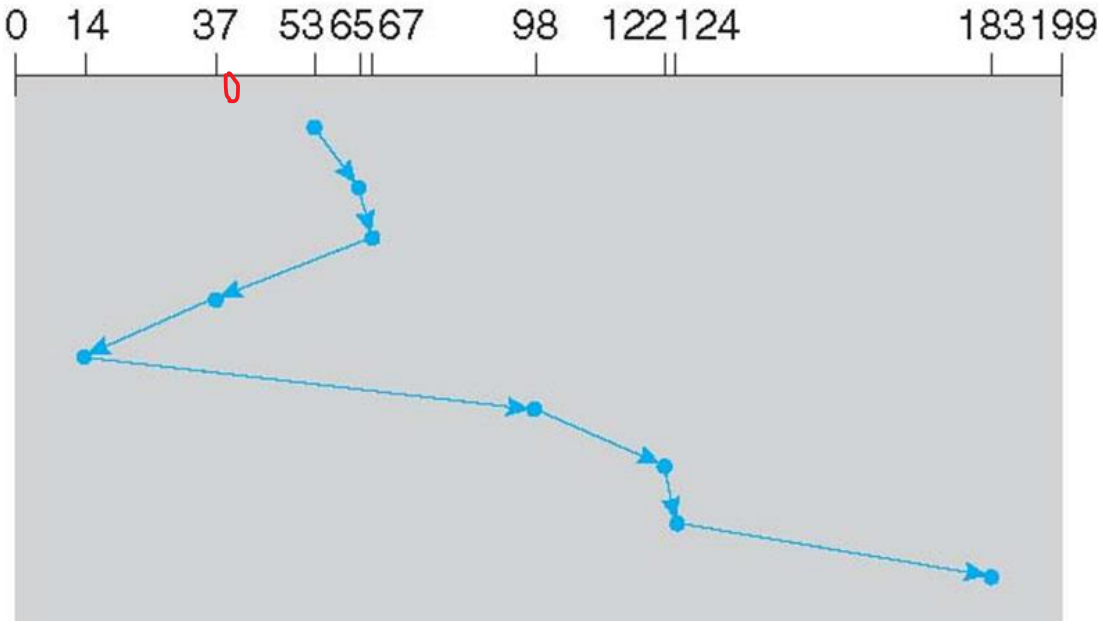
+ |65 – 67|

Shortest Seek Time First SSTF

5 8 3 6 4 7 1 2

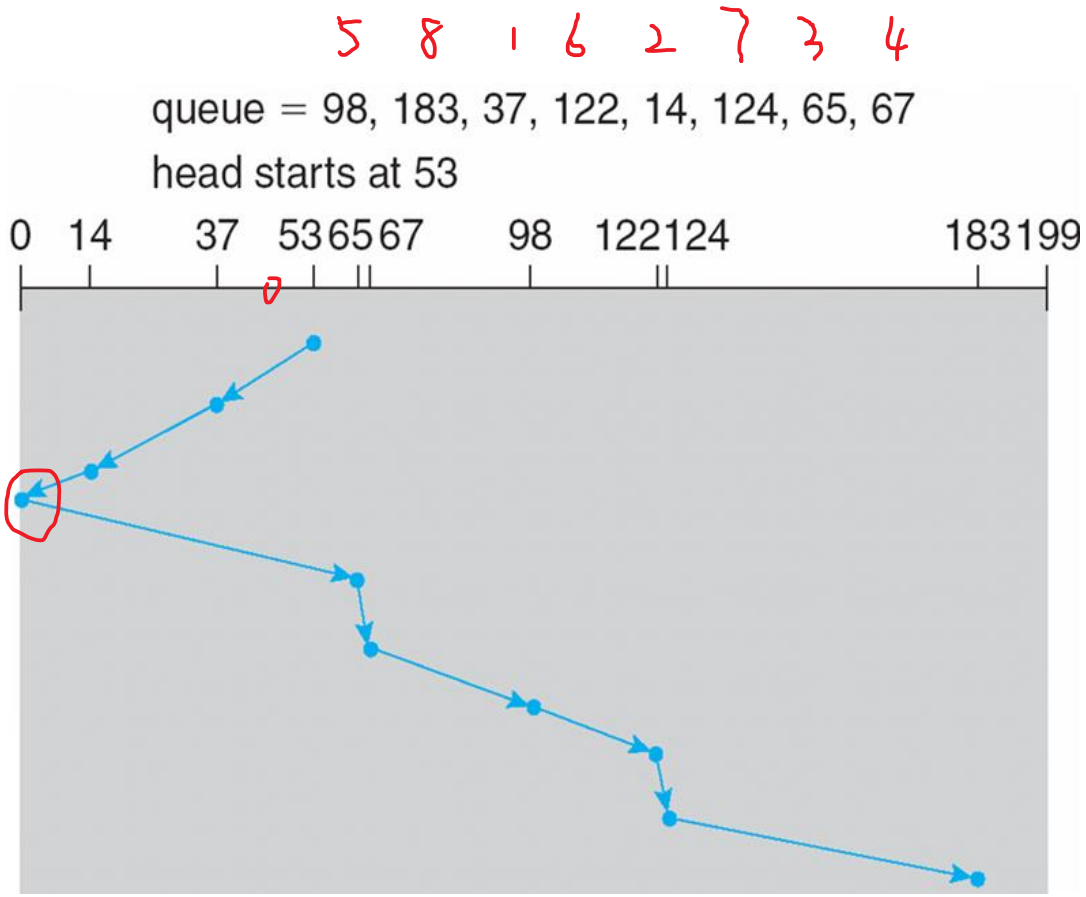
queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53



- selects the request with the minimum seek time (**the next shortest distance**) from the current head position.
- total head movement of 236 cylinders

SCAN (Elevator)



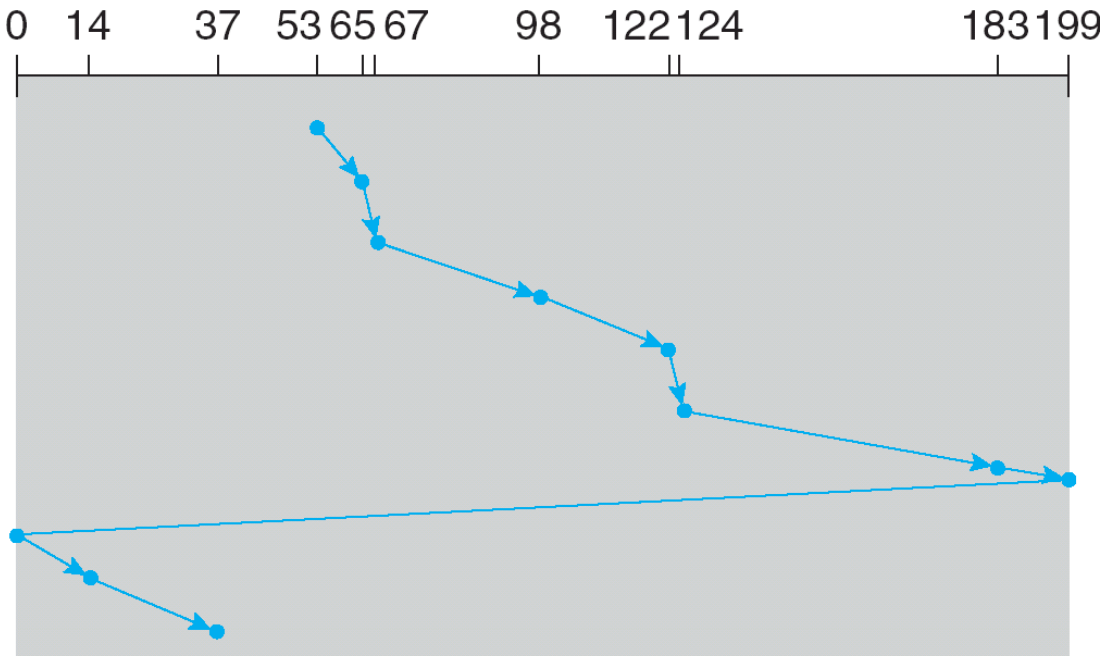
- The disk arm starts at one end of the disk (the last track), and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed, and servicing continues.
- total head movement of 236 cylinders

Circular-SCAN / C-SCAN

3 6 8 4 7 5 1 2

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

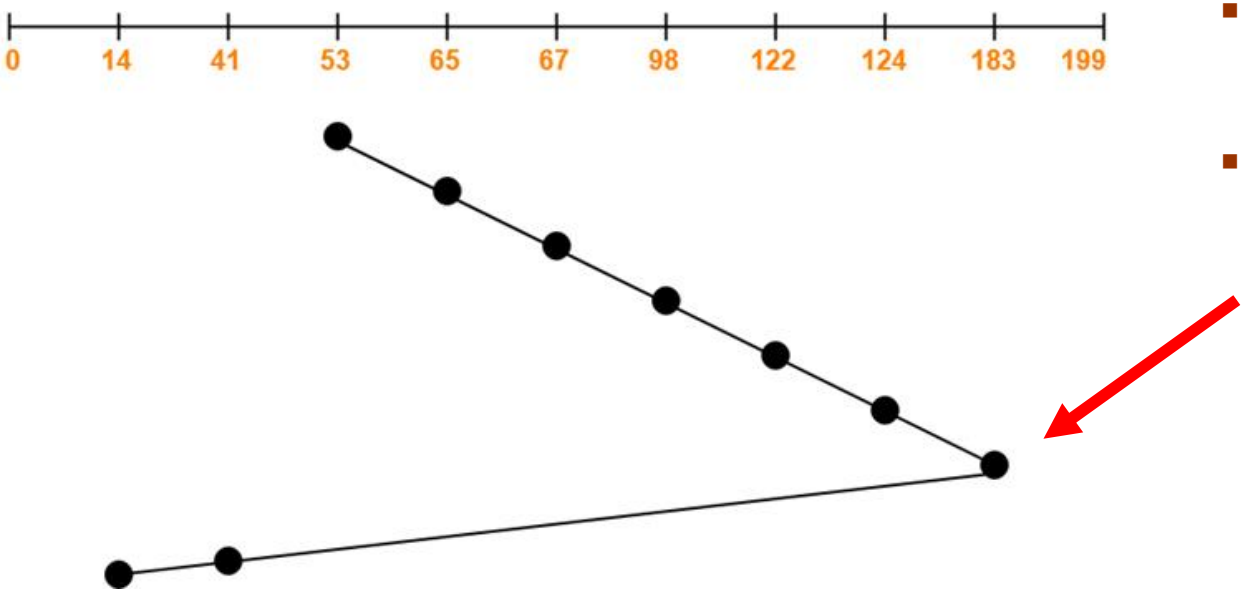


- the head moves from one end of the disk to the other, servicing requests as it goes.
- when it reaches the other end, it immediately returns to the beginning of the disk, without servicing any requests on the return trip.
- the arm is returned to the opposite end of the disk and the scan begins again.

- provides a reduced response time compared to SCAN for pending jobs on the other side of the disk.

LOOK

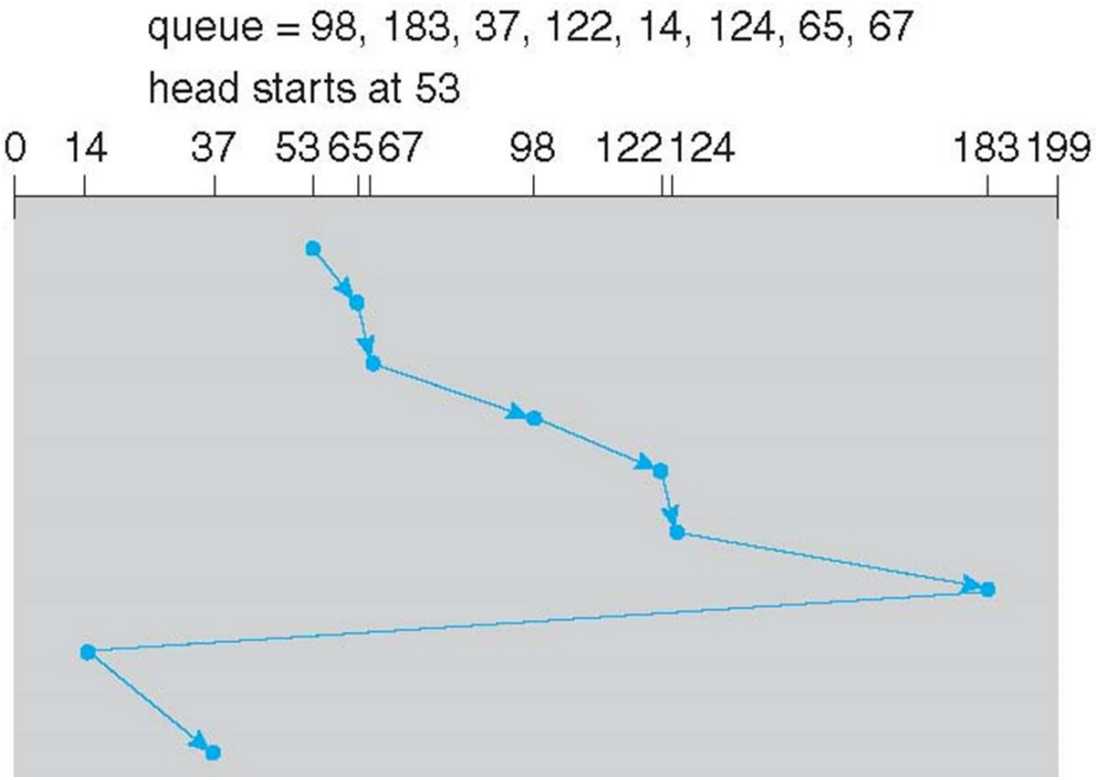
98, 183, 41, 122, 14, 124, 65, 67



- It is a variation of SCAN.
- the head moves in one direction only till the last request in that direction and reverses its direction.

- it eliminates unnecessary seek operations.

C-LOOK



- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk
- The head moves till the last request instead of last cylinder.

C-LOOK treats the request queue as circular.

- provides a reduced response time compared to LOOK for pending jobs on the other side of the disk.

Selecting a Disk-Scheduling Algorithm

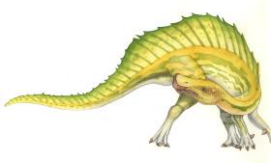
A good disk scheduling algorithm should provide:

- Minimum head/arm movement
- Minimum seek time

- **FCFS** works well with light loads; but as soon as the load grows, service time becomes unacceptably long.
- **SSTF** is quite popular and intuitively appealing. It works well with moderate loads but has the problem of localization under heavy loads.
- **SCAN** works well with light to moderate loads and eliminates the problem of indefinite postponement. SCAN is similar to SSTF in throughput and mean service times.
- **C-SCAN** works well with moderate to heavy loads and has a very small variance in service times.

Mass-Storage Systems

- Overview
- Disk Organization/Management
- Disk Attachment
- Disk Scheduling
- **Swap-Space Management**
- RAID Structure



Swap-Space Management

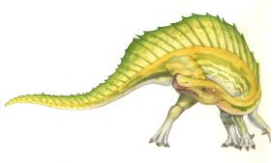
Swap-space — Virtual memory (VM) uses disk space as an extension of main memory.

How much space must be reserved for swap space?

- it depends on how much VM is needed to support the OS.
- the use of swap space may decrease the performance of the system.

Where should the swap space be allocated?

- a fast local disk is chosen for swap space, if available.
- it may be implemented out of the existing file system or in a separate disk partition.



Swap-Space Management

What will happen if more swap space is required?

- the *swap space* is implemented as a large file known as **swap file**.

- swap-file implementation may be slow (need to access the directory structure and other data structures related to the file system).

- implementation of swap files varies with the OS.

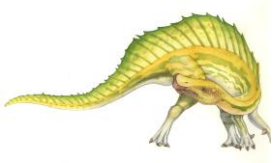
- Example: a swap file is created using *mkfile* command in some OSs.

- when changes in system configuration(s) or installation of new software packages → *more swap space* (repartitioning or another disk may be added)

- **Swap-space management**
  - Kernel uses **swap maps** to track swap-space use.

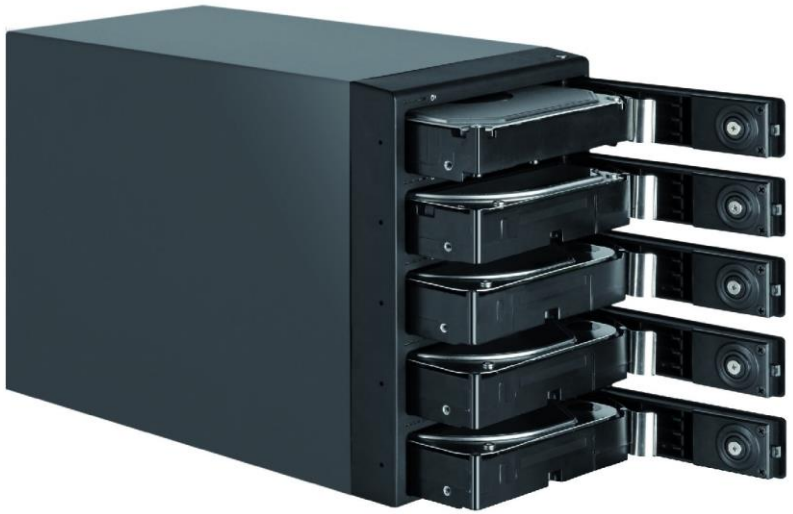
Mass-Storage Systems

- Overview
- Disk Organization/Management
- Disk Attachment
- Disk Scheduling
- Swap-Space Management
- **RAID Structure**



RAID Structure

- **RAID**: **R**edundant **A**rrays of **I**ndependent **D**isks.
- RAID is a system of **data storage that uses multiple hard disk drives to store data.**
- RAID is a set of physical drives viewed by **the operating system as a single logical drive.**
- There are several different **storage methods**, named **levels**.
- ***RAID controller*** is used for controlling a RAID array. It may be hardware- or software-based.



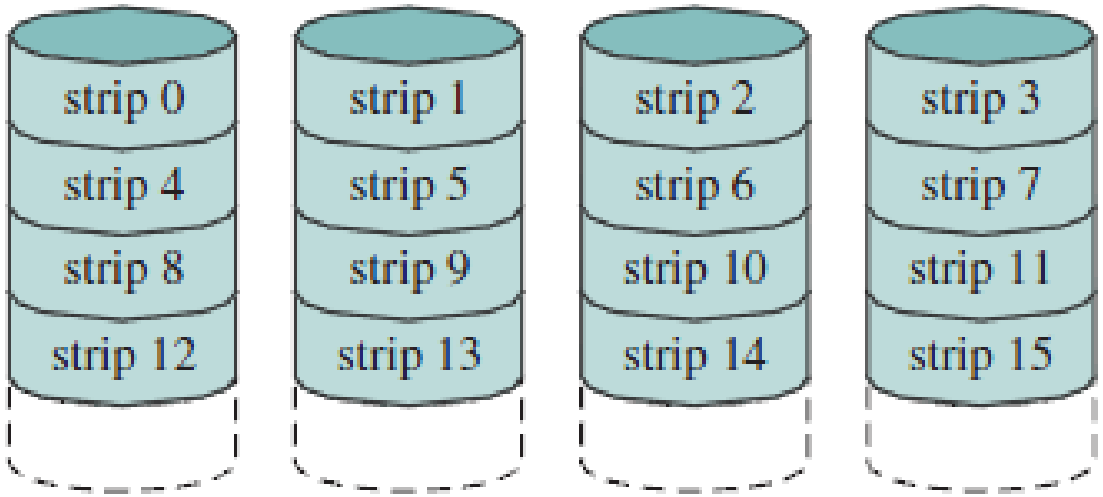
RAID

RAID uses **3 main techniques**:

- **Mirroring** is copying data to more than one drive.
  - If one disk fails, the mirror image **preserves the data from the failed disk.**
- **Striping** breaks data into “chunks” that are written in succession to different disks.
  - Striping provides **high data-transfer rates**, this improves performance because your computer can access data from more than one disk simultaneously.
- **Error correction** redundant data is stored, allowing detection and possibly fixing of errors.

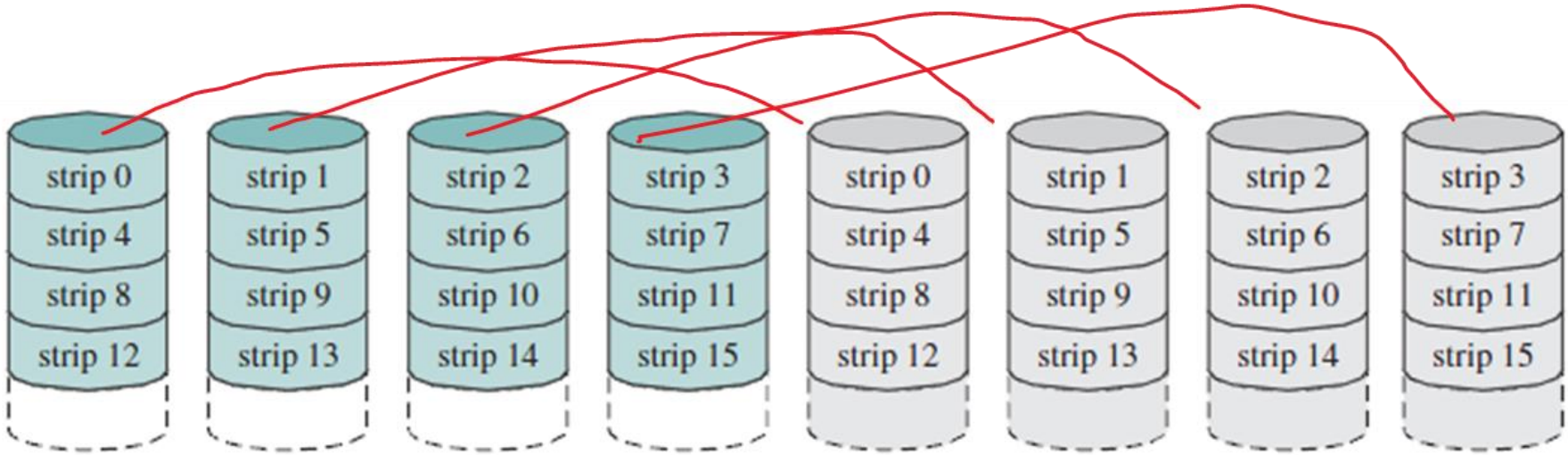
RAID Level 0

- ***disk striping***
- the data is distributed across all the disks in the form of **strips**.
- Raid 0 treats multiple disks as a single partition. The RAID technology used the term *strip* rather than the term *block*.
- RAID 0 does not provide any redundancy**.** If one of the disks in the array is not accessible, then the strips in that disk are lost.



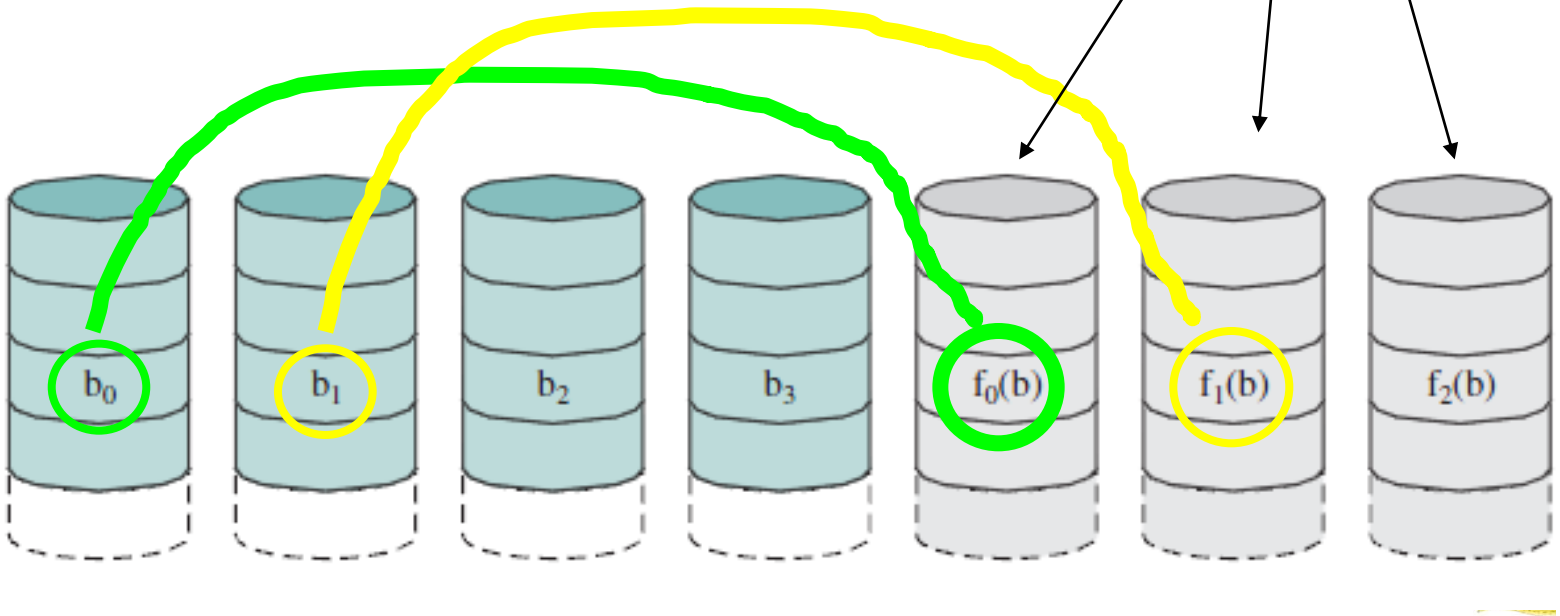
RAID Level 1

- ***disk mirroring***
- **this structure includes striping as well as redundancy.**
- is called a *mirrored* configuration because it provides redundancy by having a duplicate set of all data in a mirror array of disks (backup system in case of hardware failure).
- *write* operation on various strips can be done in parallel.



RAID Level 2

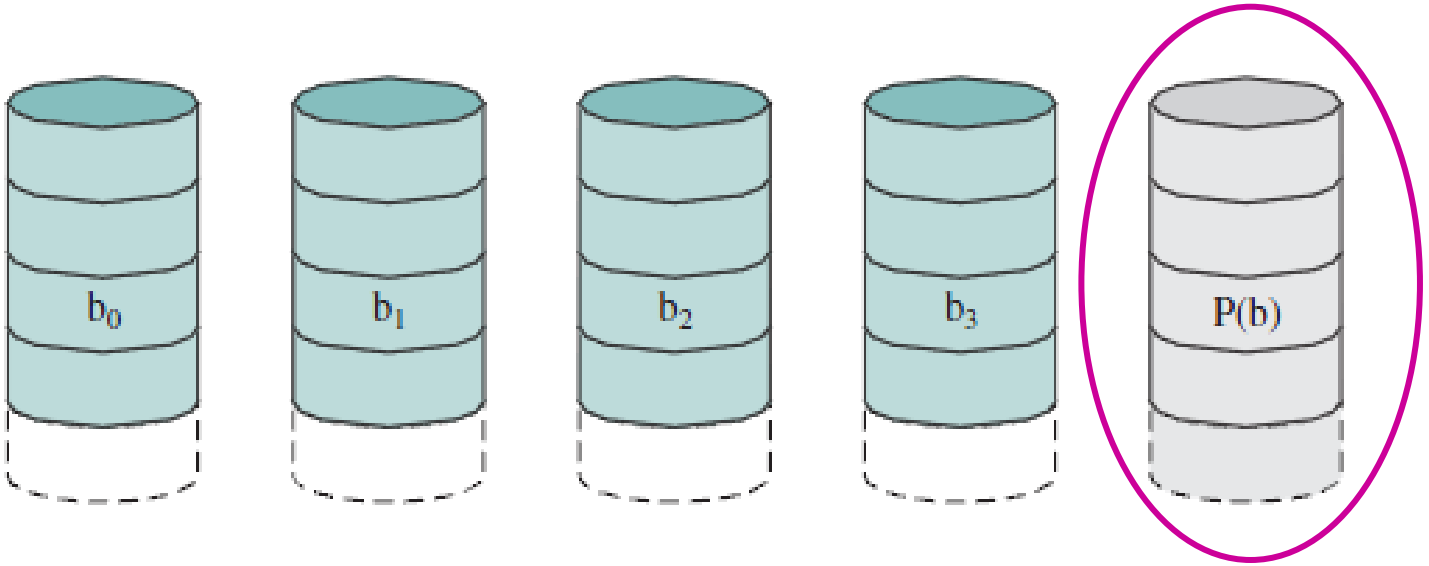
- ***memory-style error-correcting-code organization***
- an error detection and correction code (Hamming ECC code) is calculated across corresponding bits on **each disk**,
- the parity bits of the Hamming code are stored in *separate disks*.
- this structure reduces data storage.
- uses *very small strips* (as small as a single byte or word)



RAID Level 3

- ***bit-interleaved parity organization*** (a modification of level 2)
- uses *very small strips* (as small as a single byte or word).
- compute the parity of each strip (uses XOR ECC instead Hamming ECC).
- uses only single-parity disk - **dedicated parity disk** to store the parity of strips.

Suppose, strip X = {1010}, **the parity bit is 0 as there are even number of 1s.**
Suppose strip X = {1110}, **the parity bit here is 1 as there are odd number of 1s.**

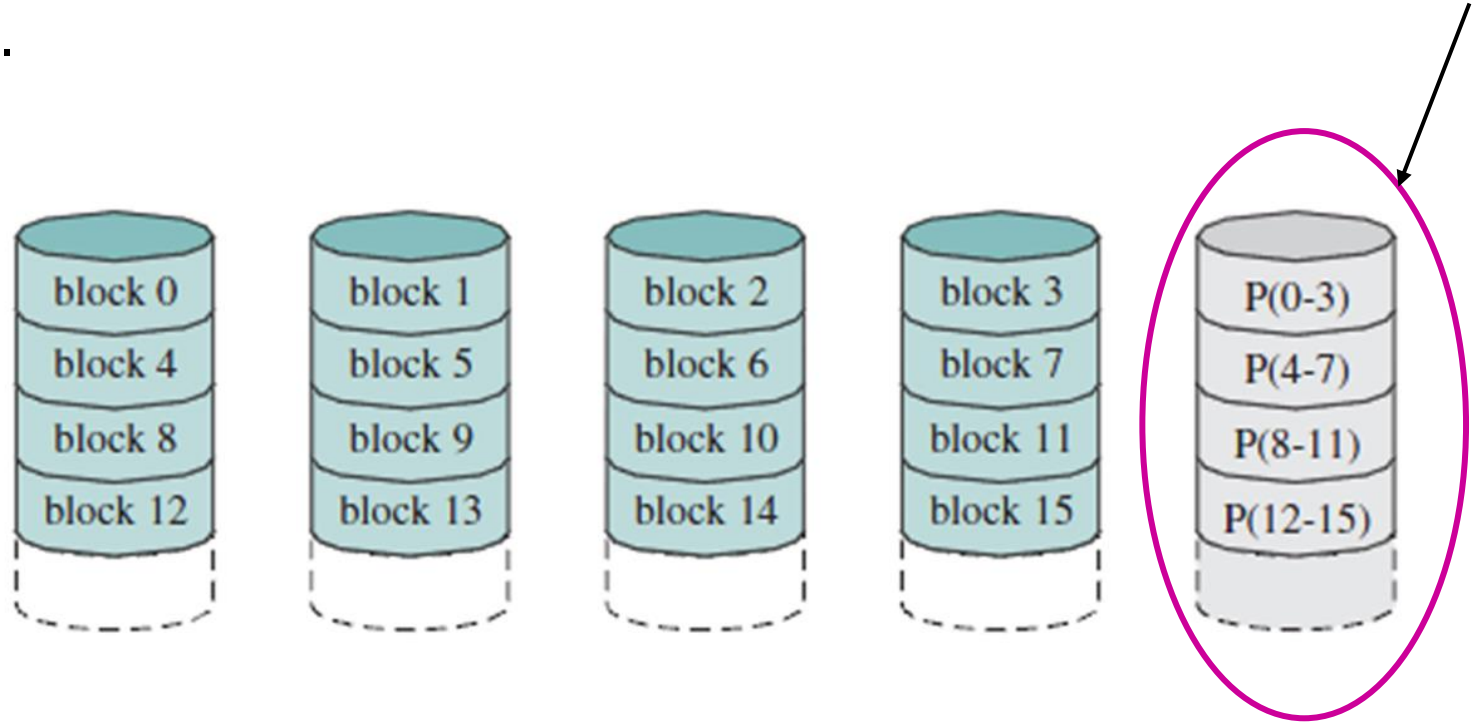


RAID Level 4

- ***block-interleaved parity organization***

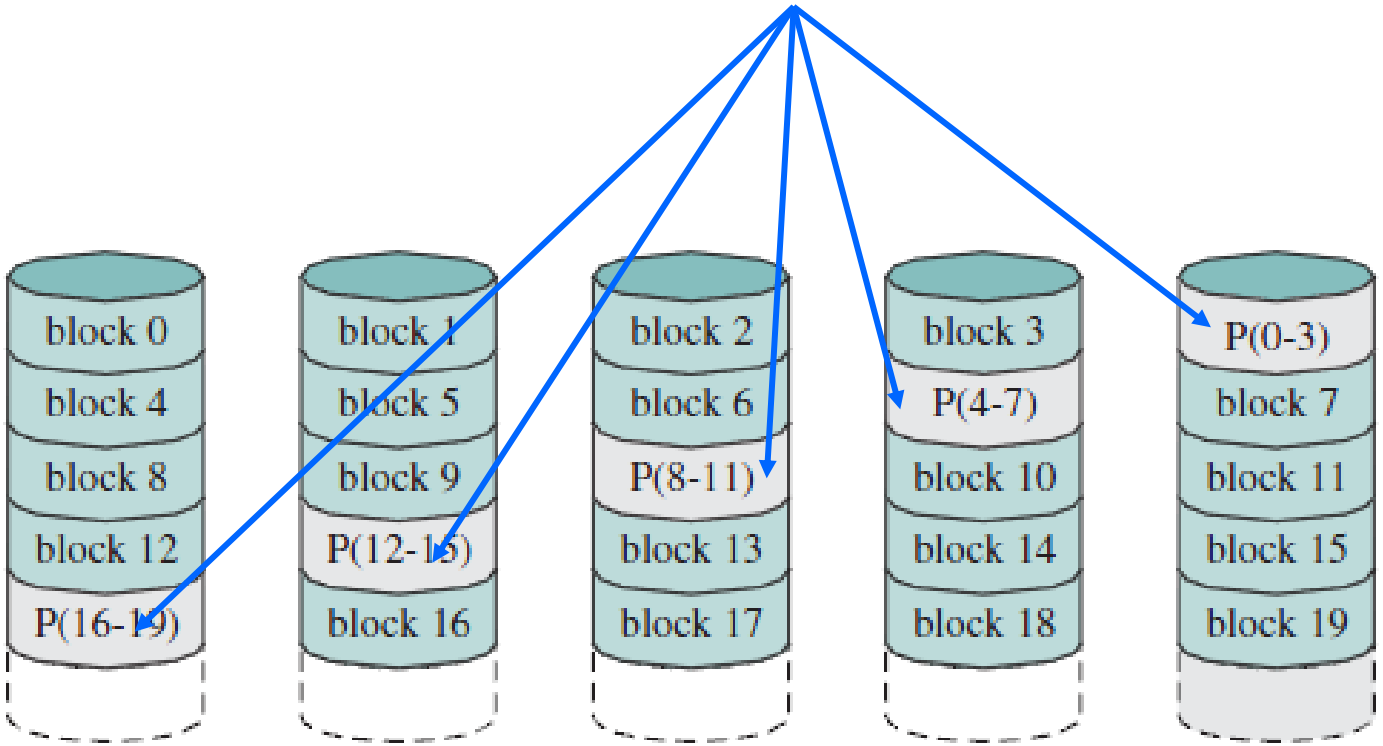
-a small data read or write access may become slow due to parity bit calculation → uses ***large-sized strips***, and the data is striped as **fixed-sized blocks** (block size is 512 bytes).

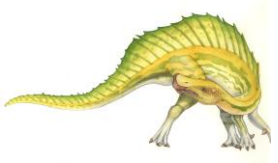
-provides **block-level striping** and stores a parity block on a **dedicated disk**.



RAID Level 5

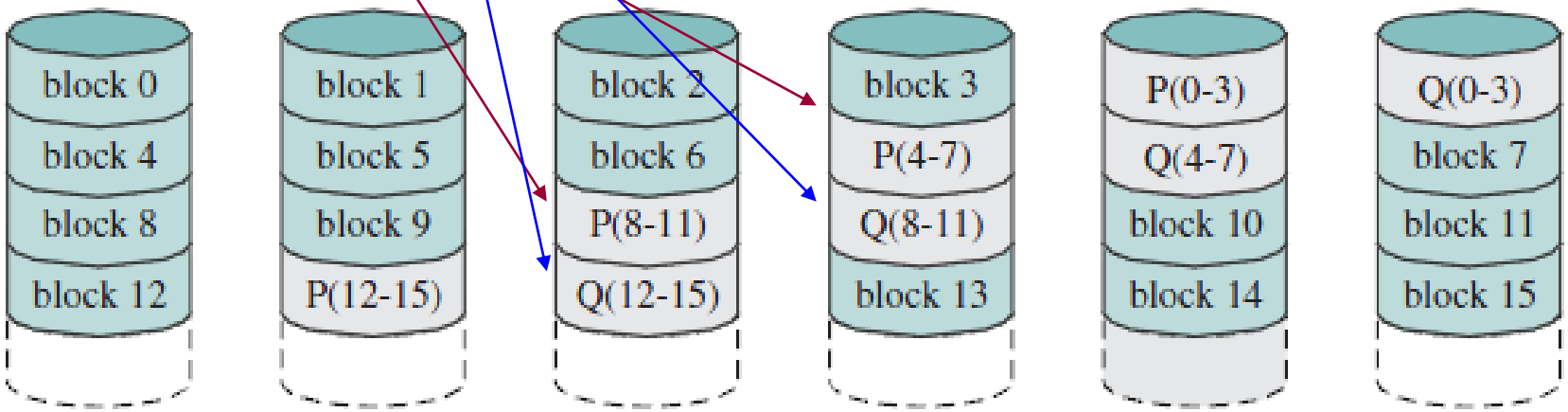
- ***block-interleaved distributed parity*** (very much like level 4).
- removes the problem of excessive use of the parity drive
- the parity bits are not stored in a single disk,
- **distributes the parity strips across the disks**.





RAID Level 6

- ***block level dual parity*** - two types of parity calculations, extra degree of *error detection and correction* (parity check and Reed-Solomon codes).
  - **P** is a **parity check** (the same as in *levels 4* and *5*)
  - **Q** is an **independent data-check algorithm**
- Both parities (P and Q) are distributed on separate disks across the array.
- The double parity allows for data restoration even if two disks fail.



Better features → combined RAID levels

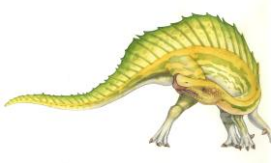
- RAID levels 0 and 1 have been combined to obtain

RAID 0 + 1

- RAID levels 1 and 0 have been combined to obtain

RAID 1 + 0

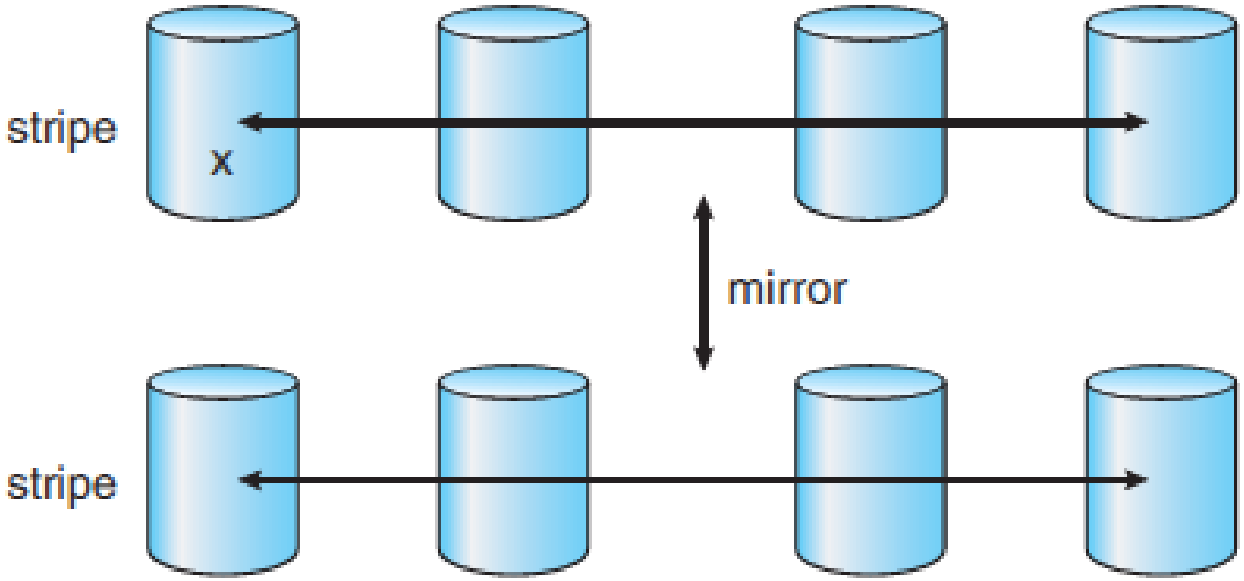
Other RAID levels such as 0 + 3, 0 + 5, and 1 + 5 have also been developed.



RAID level 0 + 1

Mirror of stripes – combinations **striping** of **RAID 0** (stripes are created) and **mirroring** of **RAID 1** (mirror is created over them).

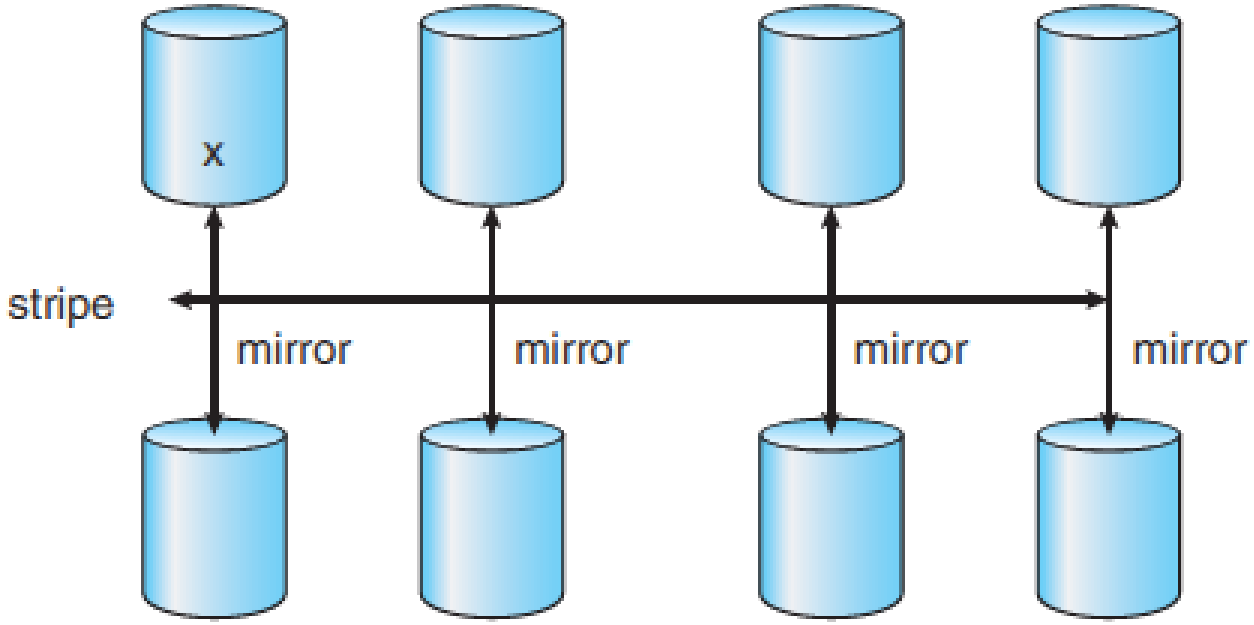
- a set of *n* disks are striped, and then the stripe is mirrored on *n* redundant disks.



RAID level 1 + 0 (a.k.a. RAID 10)

Stripe of mirrors - combines **mirroring** of **RAID 1** (mirrored drive pairs are created) with **striping** of **RAID 0** (stripe is created over these).

- in case of failure of a single disk, the mirror copy of the whole disk is available
- highly utilized any server that's performing many write operations.
- best performance, but it is also costly (requires twice as many disks as other RAID levels).



RAID Conclusions

- RAID **improves the performance** (multiple disks can be accessed in parallel).
- RAID **improves the reliability** by putting redundant copies of the data (mirroring data).
- RAID **is fast** because data can be read and written to different disks simultaneously.
- **RAID is not a backup**!!! A backup is a copy of data, which is stored somewhere else and is detached from the original data both in space and time.

End of Lecture

- **Summary**
  - Overview
  - Disk Organization/Management
  - Disk Attachment
  - Disk Scheduling
  - Swap-Space Management
  - RAID Structure

- **Reading**
  - Textbook 9th edition, **chapter 10 of the module textbook**