

OPERATING SYSTEMS

Storage Management

Chandravva Hebbi

Department of Computer Science

OPERATING SYSTEMS

Mass-Storage Structure – Swap Space and RAID

Chandravva Hebbi

Department of Computer Science

- The slides/diagrams in this course are an **adaptation**, **combination**, and **enhancement** of material from the following resources and persons:
1. Slides of Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne - 9th edition 2013 and some slides from 10th edition 2018
 2. Some conceptual text and diagram from Operating Systems - Internals and Design Principles, William Stallings, 9th edition 2018
 3. Some presentation transcripts from A. Frank – P. Weisberg
 4. Some conceptual text from Operating Systems: Three Easy Pieces, Remzi Arpaci-Dusseau, Andrea Arpaci Dusseau

- ❑ Swap-space — Virtual memory uses disk space as an extension of main memory
 - ❑ Less common now due to memory capacity increases
- ❑ Swap-space can be carved out of the normal file system, or, more commonly, it can be in a separate disk partition (raw)
- ❑ Swap-space management
 - ❑ 4.3BSD allocates swap space when process starts; holds text segment (the program) and data segment
 - ❑ Kernel uses **swap maps** to track swap-space use
 - ❑ Some systems allow the use of multiple swap spaces – both files and dedicated swap partitions

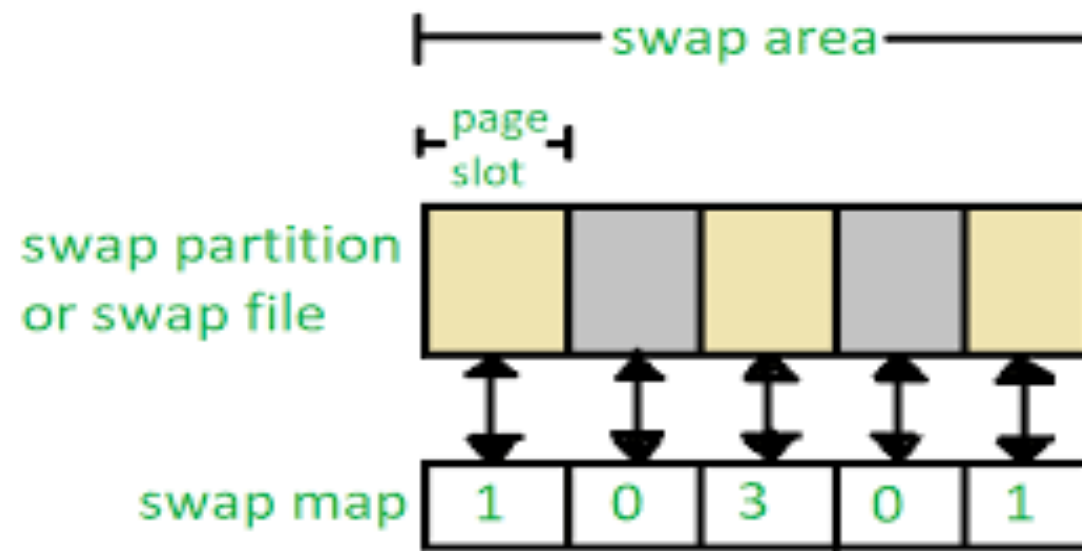
❓ Solaris 2 allocates swap space only when a dirty page is forced out of physical memory, not when the virtual memory page is first created

- ▶ File data written to swap space until write to file system requested
- ▶ Other dirty pages go to swap space due to no other home
- ▶ Text segment pages thrown out and reread from the file system as needed

❓ What if a system runs out of swap space?

❓ Some systems allow multiple swap spaces

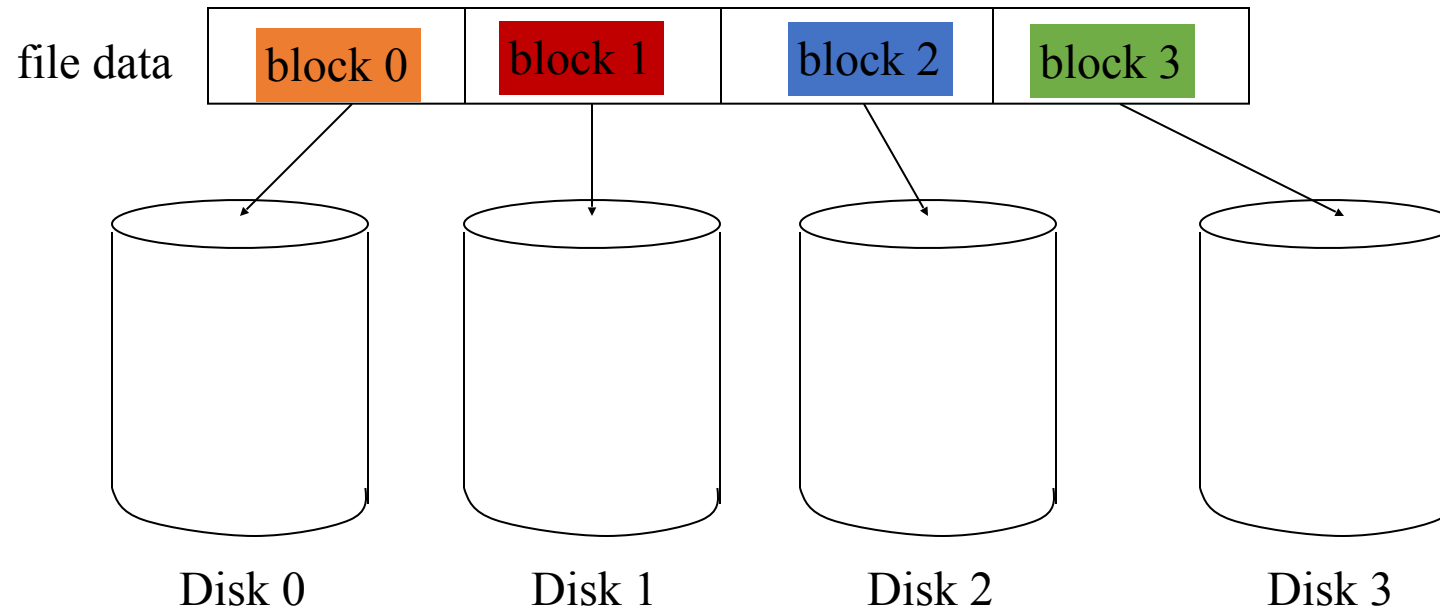
- ❑ Linux allows one or more swap areas to be established.
- ❑ A swap area may be in either a swap file on a regular file system or a dedicated swap partition.
- ❑ Each swap area consists of a series of 4-KB **page slots**, which are used to hold swapped pages.
- ❑ Associated with each swap area is a **swap map**—an array of integer counters, each corresponding to a page slot in the swap area.
 - ❑ 0 => page slot is available
 - ❑ 3 => swapped page is mapped to 3 different processes



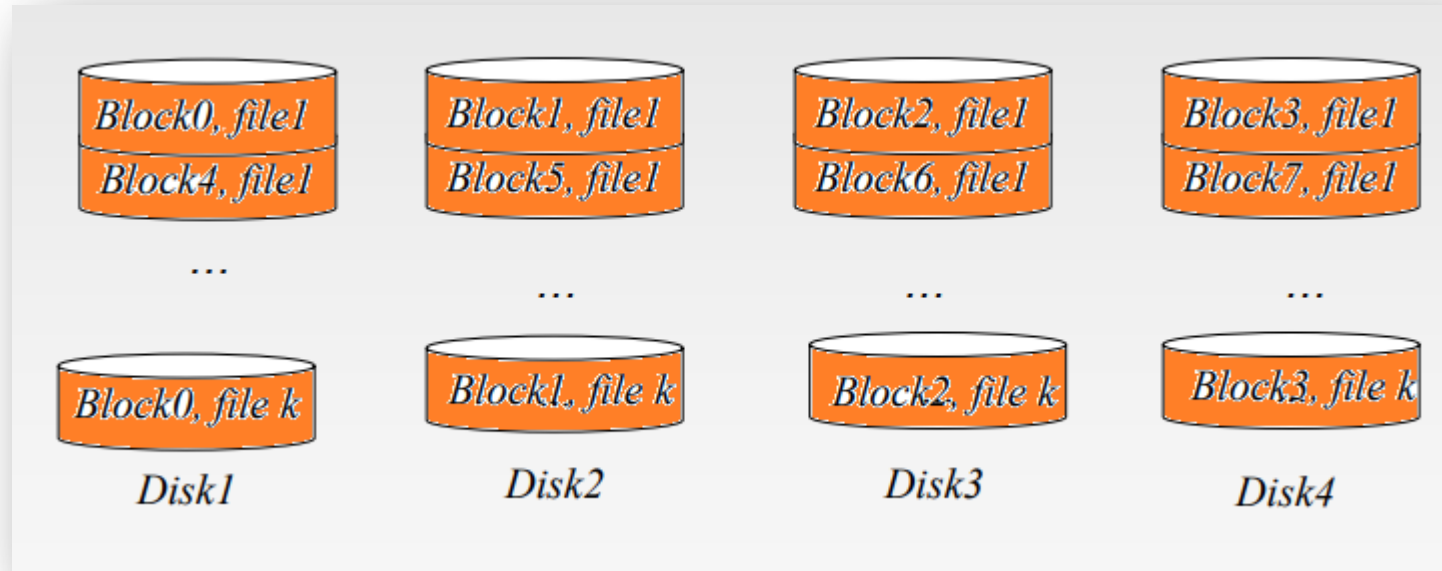
- ❑ RAID is a Disk Organization technique
 - Addresses performance and reliability issues
 - Multiple disk drives provide (or improve) reliability via **redundancy**
- ❑ Many systems today need to store many terabytes of data
- ❑ Don't want to use single, large disk
 - too expensive
 - failures could be catastrophic
- ❑ Would prefer to use many smaller disks
- ❑ **Redundant Array of Independent (inexpensive) Disks**

- ❑ Basic idea is to connect multiple disks together to provide
 - ❑ large storage capacity
 - ❑ faster access to reading data
 - ❑ redundant data
- ❑ Many different levels of RAID systems
 - ❑ differing levels of redundancy, error checking, capacity, and cost

- Take file data and map it to different disks
- Allows for reading data in parallel
- Transfer rate is improved by striping data across the disks
 - **Bit-level striping and block-level striping (most common)**



- ❑ With n disks, block i of a file goes to disk $(i \bmod n) + 1$
 - Requests for different blocks can run in parallel if the blocks reside on different disks
 - A request for a long sequence of blocks can utilize all disks in parallel



- Keep two copies of data on two separate disks
- Gives good error recovery
 - if some data is lost, get it from the other source
- Expensive
 - requires twice as many disks
- Write performance can be slow
 - have to write data to two different spots
- Read performance is enhanced
 - can read data from file in parallel

? Mean time to failure

? If MTTF of a single disk is 100,000 hours, MTTF of some disk in an array of 100 disks = $100000/100 = 1000$ hours or 41.66 days

? **Mean time to repair** – time taken to replace a failed disk and to restore the data on it

? **Mean time to data loss** based on above factors

? If mirrored disks fail independently (i.e., not related to power failures and natural disasters), consider disk with MTTF of 100,000 hours and MTTR is 10 hours

? Mean time to data loss of a mirrored disk system is $100,000^2 / (2 * 10) = 500 * 10^6$ hours, or 57,000 years!

- ❑ Frequently combined with **NVRAM** to improve write performance
- ❑ Several improvements in disk-use techniques involve the use of multiple disks working cooperatively
- ❑ Disk **striping** uses a group of disks as one storage unit
- ❑ RAID is arranged into six different levels
- ❑ RAID schemes improve performance and improve the reliability of the storage system by storing redundant data
 - ❑ **Mirroring** or **shadowing** (**RAID 1**) keeps duplicate of each disk
 - ❑ Striped mirrors (**RAID 1+0**) or mirrored stripes (**RAID 0+1**) provides high performance and high reliability
 - ❑ **Block interleaved parity** (**RAID 4, 5, 6**) uses much less redundancy

- ❑ RAID within a storage array can still fail if the array fails, so automatic **replication** of the data between arrays is common
- ❑ Frequently, a small number of **hot-spare** disks are left unallocated, automatically replacing a failed disk and having data rebuilt onto them



(a) RAID 0: non-redundant striping.



(b) RAID 1: mirrored disks.



(c) RAID 2: memory-style error-correcting codes.



(d) RAID 3: bit-interleaved parity.



(e) RAID 4: block-interleaved parity.



(f) RAID 5: block-interleaved distributed parity.



(g) RAID 6: P + Q redundancy.

P indicates error-correcting bits and C indicates a second copy of the data

P + Q redundancy scheme uses 2 parity values P and Q

❑ **RAID level 0** refers to disk arrays with striping at the level of blocks

- No redundancy (such as mirroring or parity bits)
- lots of disks means low Mean Time To Failure (MTTF)



(a) RAID 0: non-redundant striping.

❑ **RAID level 1** refers to disk mirroring.

- A complete file is stored on a single disk
- A second disk contains an exact copy of the file
- Provides complete redundancy of data
- Read performance can be improved
- file data can be read in parallel
- Write performance suffers
- must write the data out twice
- Most expensive RAID implementation
- requires twice as much storage space



(b) RAID 1: mirrored disks.

❑ **RAID level 2** is also known as memory-style error correcting code (ECC) organization.

❑ Stripes data across disks similar to Level-0

- difference is data is **bit** interleaved instead of **block** interleaved
- For ex, the first bit of each byte can be stored in disk 1, the second bit in disk 2, and so on until the eighth bit is stored in disk 8; the error-correction bits are stored in further disks.

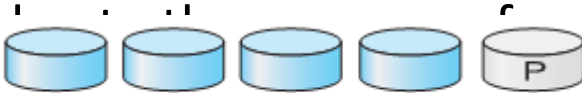
❑ Uses ECC to monitor correctness of information on disk

- Multiple disks record the ECC information to determine which disk is in fault
- A parity disk is then used to reconstruct corrupted or lost data

❑ RAID level 2 requires only 3 disks' overhead for 4 disks of data, unlike RAID level 1, which requires 4 disks' overhead.



(c) RAID 2: memory-style error-correcting codes.

- ❑ **RAID level 3**, or bit-interleaved parity organization;
 - ❑ One big problem with Level-2 is the number of extra disks needed to detect which disk had an error
 - ❑ Modern disks can already determine if there is an error
 - using ECC codes with each sector
 - ❑ So just need to include a parity disk
 - if a sector is bad, the disk itself tells us, and use the parity disk to correct it
 - ❑ Transfer rate for reading or writing a single block is faster than RAID level 1.
 - ❑ But supports fewer I/Os per second, since every disk has to participate in every I/O request.
 - ❑ Has performance problem  computing and writing the parity.



❑ RAID level 4 interleaves file blocks

- allows multiple small I/O's to be done at once



(e) RAID 4: block-interleaved parity.

- ❑ Consists of block-level striping with dedicated parity.
- ❑ Still use a single disk for parity
- ❑ Now the parity is calculated over data from multiple blocks
 - Level-2,3 calculate it over a single block
- ❑ If an error detected, need to read other blocks on other disks to reconstruct data
 - Doing multiple small reads is now faster than before
 - However, writes are still very slow
 - this is because of calculating and writing the parity blocks
 - Also, only one write is allowed at a time
 - all writes must access the check disk so other writes have to wait

- ❑ **RAID level 5** stripes file data and checks data over all the disks

- no longer a single check disk
- no more write bottleneck



(f) RAID 5: block-interleaved distributed parity.

- ❑ Consists of block-level striping with distributed parity.

- Unlike RAID 4, parity information is distributed among the drives

- ❑ Drastically improves the performance of multiple writes

- they can now be done in parallel

- ❑ Slightly improves reads

- one more disk to use for reading

- ❑ read and write performance close to that of RAID Level-1

- ❑ requires as much disk space as Levels-3,4

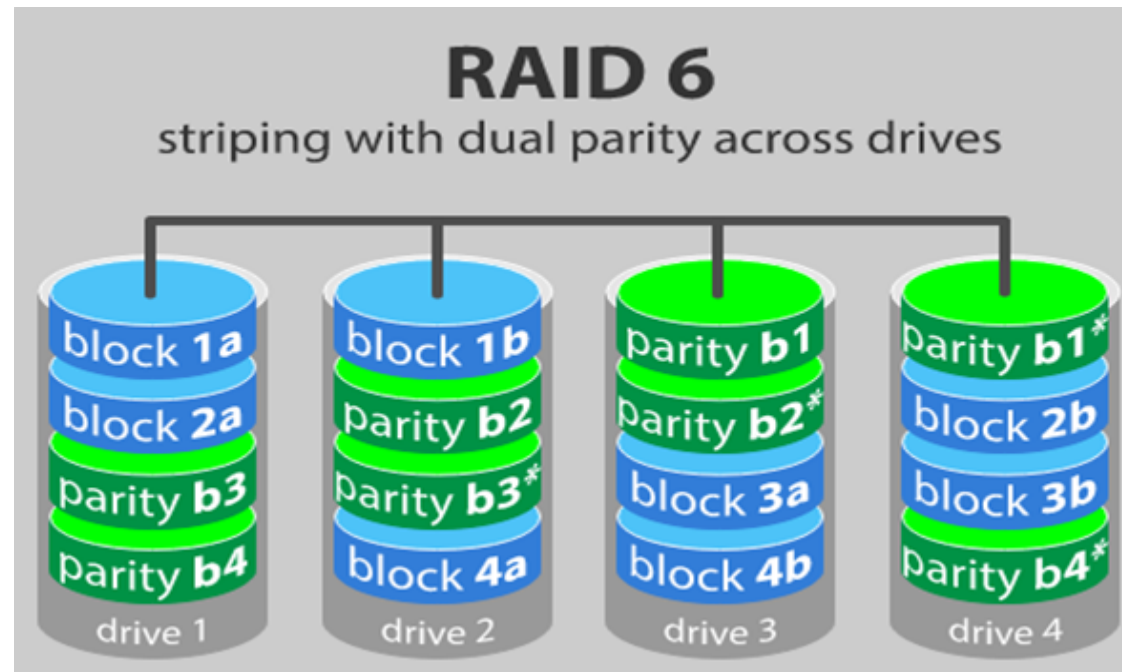
OPERATING SYSTEMS

RAID Level-6

- ❑ RAID 6 is like RAID 5, but the parity data are written to two drives.
 - That means it requires at least 4 drives and can withstand 2 drives dying simultaneously.
- ❑ Write data transactions are slower than RAID 5 due to the additional parity data that have to be calculated.



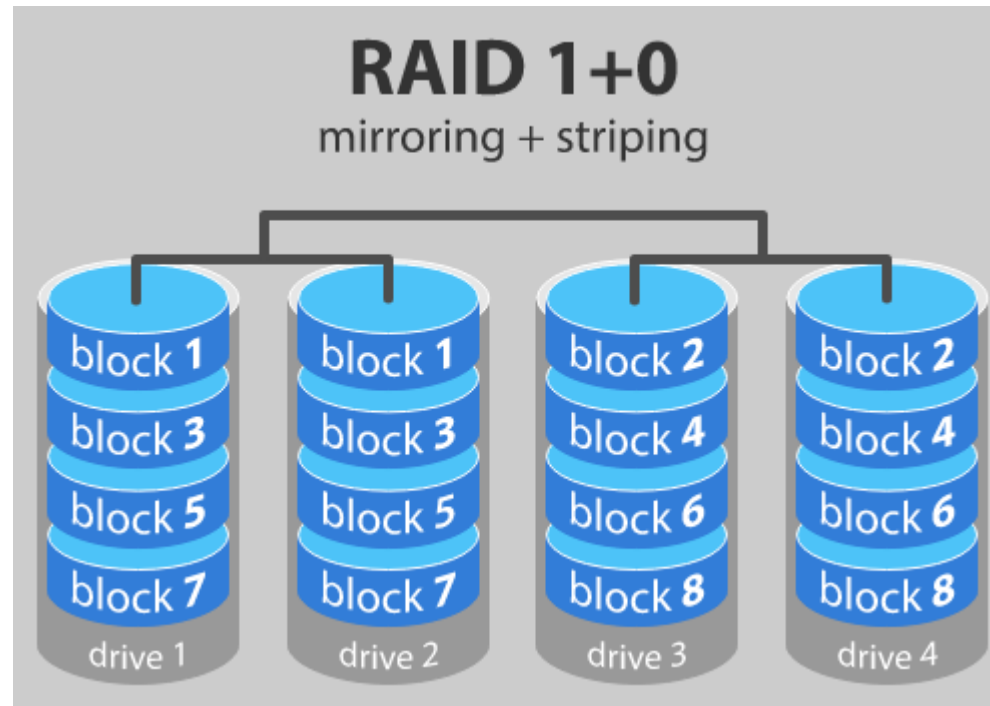
(g) RAID 6: P + Q redundancy.



OPERATING SYSTEMS

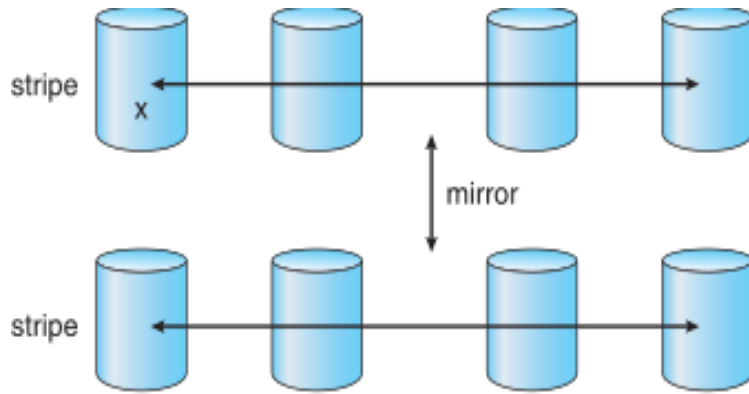
RAID level 10 - combining RAID 1 and RAID 0

- ❑ Provides security by mirroring all data on secondary drives while using striping across each set of drives to speed up data transfers.
- ❑ Rebuild time is very fast
- ❑ Half of the storage capacity goes to mirroring
 - so compared to large RAID 5 or RAID 6 arrays, this is an expensive way to have redundancy.



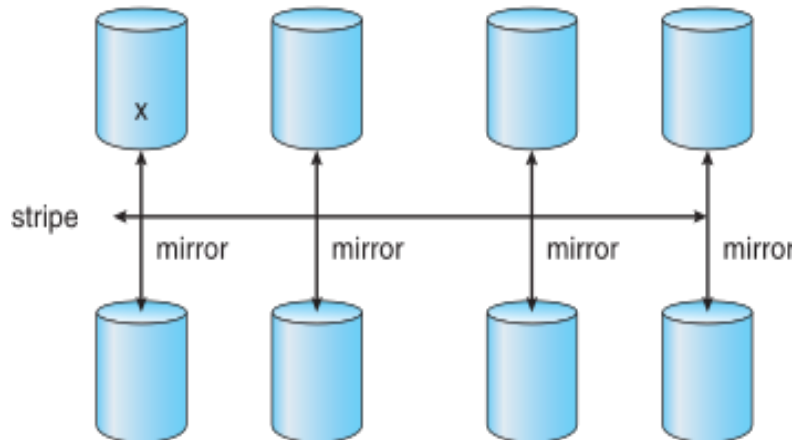
OPERATING SYSTEMS

RAID (0 + 1) and (1 + 0)



a) RAID 0 + 1 with a single disk failure.

Disks are striped and then the stripe is mirrored to another, equivalent stripe. If a single disk fails, an entire stripe is inaccessible.



b) RAID 1 + 0 with a single disk failure.

Disks are mirrored in pairs and then the resulting mirrored pairs are striped. This is better than 0+1 when a single disk fails

- ❓ One consideration is rebuild performance.
 - ❓ If a disk fails, the time needed to rebuild its data can be significant.
 - ❓ This may be an important factor if a continuous supply of data is required, as it is in high-performance or interactive database systems.
 - ❓ Furthermore, rebuild performance influences the mean time to failure.
 - ❓ Rebuild performance varies with the RAID level used.
 - ▶ Rebuilding is easiest for RAID level 1, since data can be copied from another disk.

- ▶ For the other levels, we need to access all the other disks in the array to rebuild data in a failed disk.
- ▶ Rebuild times can be hours for RAID 5 rebuilds of large disk sets.
- ❓ RAID level 0 is used in high-performance applications where data loss is not critical.
- ❓ RAID level 1 is popular for applications that require high reliability with fast recovery.
- ❓ RAID 0 + 1 and 1 + 0 are used where both performance and reliability are important—for example, for small databases.
- ❓ Due to RAID 1's high space overhead, RAID 5 is often preferred for storing large volumes of data.

- ❑ RAID system designers and administrators of storage have to make several other decisions as well.
 - ❑ How many disks should be in a given RAID set?
 - ❑ How many bits should be protected by each parity bit?
 - ❑ If more disks are in an array, data-transfer rates are higher, but the system is more expensive.
 - ❑ If more bits are protected by a parity bit, the space overhead due to parity bits is lower, but the chance that a second disk will fail before the first failed disk is repaired is greater, and that will result in data loss.



THANK YOU

Chandravva Hebbi

Department of Computer Science Engineering

chandravvahebbi@pes.edu