

## T-Question 12.1: Storage

- a. Name two reasons why the OS has only limited information about the layout of data on hard disks (with regard to locality).

1 T-pt

**Solution:**

- Logical block addressing (LBA) abstracts away from disk layout.
- Sector sparing can cause blocks to be far apart on the disk, although their block numbers (or even CHS addresses) are close together.

- b. Why are flash-based solid-state drives (SSD) much faster than hard disks for random access?

1 T-pt

**Solution:**

*Randomly addressing flash chips / packages is faster than mechanically moving around a hard disk arm. (Further, with NCQ, requests to data in different flash chips can be handled in parallel, in contrast to a hard disk's single arm)*

- c. What technique should file systems on SSDs apply to help with write performance? How does that mechanism help the SSD to reduce write overhead?

2 T-pt

**Solution:**

*File systems should use the `trim` command to tell the SSD controller about deleted/unused blocks – in particular, when deleting files.*

*Knowledge about unused/empty blocks enables the SSD to turn these blocks into spare blocks (erased ahead of time, speeding up writing) and to avoid write amplification (copying unused blocks when re-writing a block).*

- d. Why is a RAID 5 system often preferred over a RAID 4 system?

1 T-pt

**Solution:**

*Using RAID4, every data write implies a write to the very same parity disk. When RAID5 is used, the parity read and write load is distributed among all drives.*

## T-Question 12.2: Files

- a. Sparse files have unallocated holes which are represented by zeros. When a write is performed to an offset in a hole, a new block is allocated that can hold the written data. How can you determine if a file is a sparse file (i.e., it has holes), using the `stat` system call? *Hint:* Take a look at the man-page and the fields in `struct stat`.

1 T-pt

**Solution:**

*The `struct stat` returned by the system call contains the number of allocated 512 byte blocks (`st_blocks`) as well as the file size `st_size`. By checking if `st_blocks * 512 == st_size`, we can determine if enough blocks are allocated or if there must be holes in the files.*

b. What are ACLs?

**1 T-pt**

**Solution:**

*Access control lists (ACLs) consist of access control entries (ACEs), which each assign an individual subject individual access permissions. Depending on the system and use case, a subject may be a user, a group of users, a computer (e.g., in a network).*

c. UNIX defines the **r**ead, **w**rite, and **e**xecute permissions. What other permissions on files could be of use? Name at least two.

**1 T-pt**

**Solution:**

- *Permission to delete a file*
- *Separate read data and attribute permissions (e.g., to only allow making statistics such as file sizes without permission to read data)*
- *Separate write and append data permissions (e.g., to prevent logging services from manipulating previous log entries)*
- *Permissions to regulate permission adjustment (take ownership, read permissions, change permissions, ...)*

**Total:  
8 T-pt**