Assignment: 2 (Homework_ch: 10)

① List the physical storage media available on the computers you use routinely. Give the speed with which data can be accessed on each medium?

→ Physical storage media :- Cache
Main memory
Flash memory
Magnetic disk
Optical storage

Cache - Fastest and most costly form of storage
Main memory - fast access (10s to 100s of nanoseconds; 1 nanosecond = 10^{-9} second)
Flash memory - Reads are roughly as fast as main memory, but writes are slow, erase is slower
Magnetic disk - More slower access than main memory
Optical storage - Reads & writes are slower tha magnetic disk.

② Explain why the allocation of records to blocks affects database-system performance significantly?

→ Since data are transferred between disk storage & main memory in units of a block, it is worthwhile to assign file records to blocks in such a way that a single block contains related records. If we can access several of the records we want with only one block access, we save disk access. Since disk access are usually the bottleneck in the performance of a database system, careful assignment of records to blocks can significantly improves performance.

③ List two advantages & two disadvantages of each of the following strategies for sorting a relational database.

→ a) Store each table in a seperate file:
→ Advantages :- uses the file system provided by the operating system, thus simplifying the DBMS for backups, etc.
Disadvantages :- restrict the ability of the DBMS to increase performance by using more sophisticated storage structures.

b) Sort multiple relations (perhaps even the entire database) in one file.
→ Advantages :- Complex structures can be implemented through the DBMS, thus improving performance.
Disadvantages :- increase the size & complexity of DBMS.

④ In the sequential file organization, why is an overflow block used even if there is, at the moment, only one overflow record?

→ In the sequential file organization, an overflow block is used because a block is the smallest space which can be read from a disk. Therefore, using any smaller region would not be useful from a performance standpoint. The space saved by allocating disk storage in record units would be overshdowed by the performance cost of allowing blocks to contain records of multiple files.

⑤ Standard buffer managers assume each block is of the same size and costs the same to read. Consider a buffer manager that, instead of LRU, uses the rate of reference to objects, that is how often an object has been accessed in the last n seconds. Suppose we want to store in the buffer objects of varying sizes of varrying read costs. Suggest how a buffer manager may choose which block to evict from the buffer.

→ A solution can make use of a priority queue to evict objects, where the priority (p) is define as the expected costs of reneading an objects, ~~whe~~ given its past access frequancy (f) and its re-read cost (c).

$$p = f \cdot c$$

The buffer manager should choose to evict the object with lowest expected cost of re-reading. Also, assuming a fixed-size buffer space, we might want to assign a lower priority to large objects so that the buffer can hold relatively more objects. One can extend the priority as $p = f \cdot c / s$

where $s \rightarrow$ size of the object.