# Cache and Locality:

Cache is on processor memory to fasten the execution process and reduce latency.

To perform well on a memory hierarchy algorithm, they must have high locality.

**Spatial Locality:** When a block of data is bought into the cache, it should contain as much useful data as possible.

**Temporal Locality:** Once a data point is in the cache, as much useful work should be done on it before evicting it from the cache.

# CPU-Bound vs Memory-bound Algorithms:

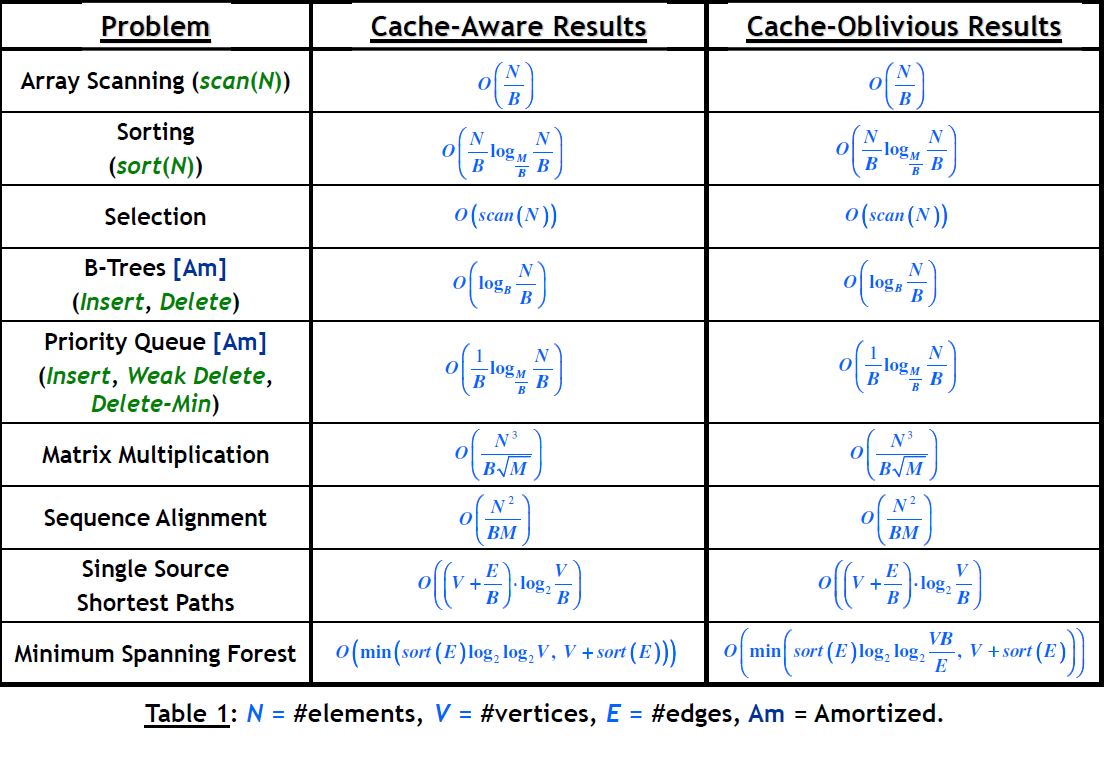
**Op-Space Ratio:** Ratio of number of operations performed by an algorithm to the amount of space (input/output) it uses.

Op-Space ratio will be high for CPU-bound and low for memory-bound.

**Two-level I/O Model:** It works on the knowledge of cache size and block size in the external memory.

Such algorithms are also called as *Cache-aware algorithms*.

**Cache Oblivious Algos :** Are built without the knowledge of M and B.



**Tall Cache:** Let Cache can hold M objects, and memory is divided into objects (each size B). If we work under an assumption that M=Ω (

# Matrix Multiplication using Caching:

**Row Order storage:** Values of a matrix are stored in a row major order. i.e. elements of a row are stored in continuous memory locations.

**Column Major Order:** Column data in sequential locations.