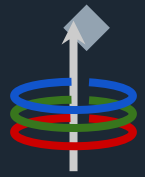


CoE 163

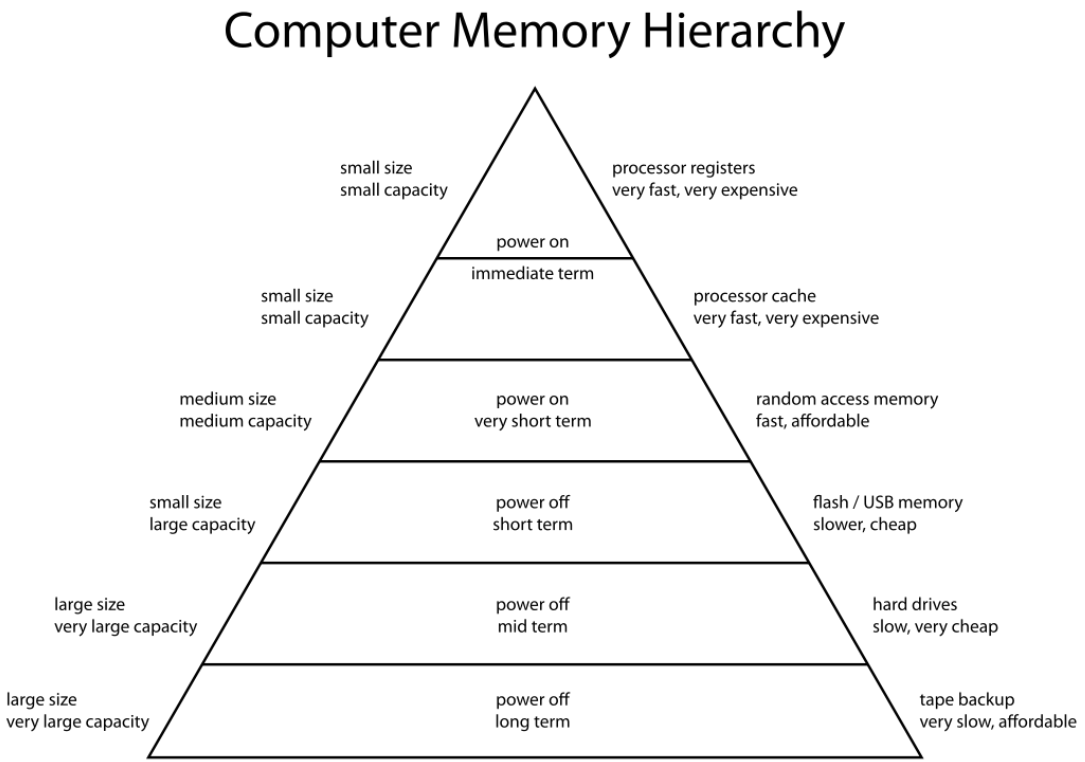
Computing Architectures and Algorithms

Review of Cache

Let’s review some things about computer memory

- Computer memories are built as hierarchies
  - As we go down hierarchy
    - Decreasing cost per bit
    - Increasing capacity
    - Increasing access time
    - Decreasing frequency of access of the memory by the processor

Let’s review some things about computer memory

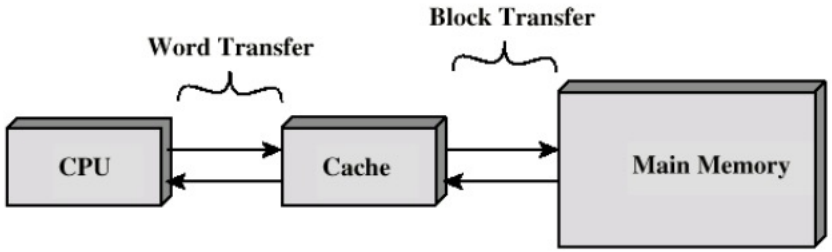


Locality of Reference

- Temporal Locality
  - Generally, we reference same memory locations at a future point in time
  - Programs often use loops and linger on the same data
- Spatial Locality
  - Programs tend to access memory locations that are near each other
  - We often store data as arrays, thus often accessing contiguous memory

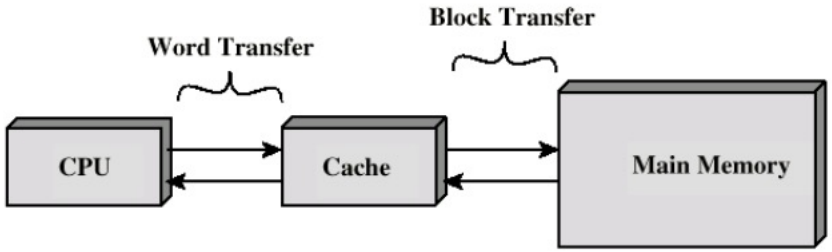
Processor Cache

- Small fast memory
- Located between normal main memory and CPU
  - May be located on CPU chip or module



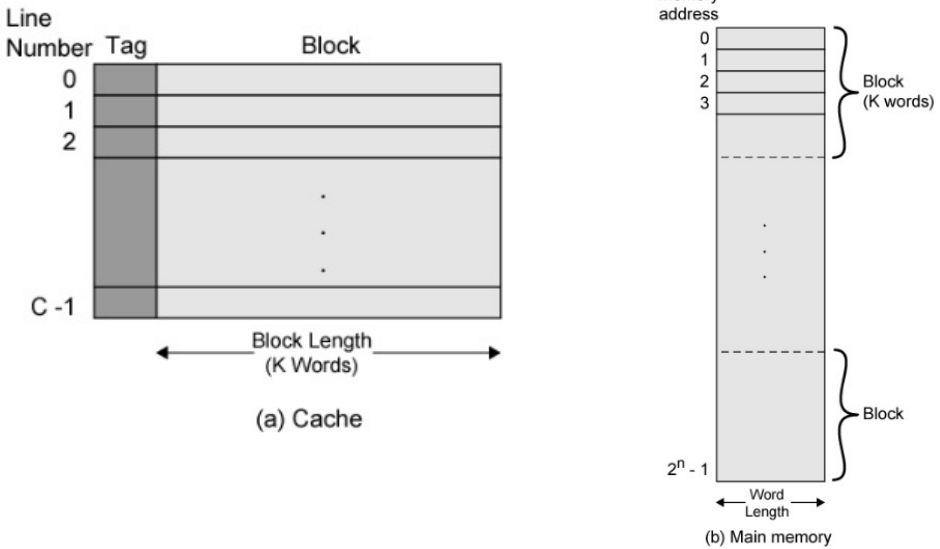
Processor Cache

- Entire blocks of data are copied from memory to the cache
  - In principle, once a byte is accessed, highly likely to access the bytes nearby (locality of reference)



Typical Cache Design

- Caches are divided into blocks
  - Number of blocks usually a power of 2



Cache behavior

- **Hit:** CPU tries to read from memory, the address is sent to cache controller and finds that the data contained by that address is in the cache
- **Miss:** data is not yet in cache; needs to be copied from main memory
- Writing to cache
  - Replacement policy (such as LRU) determines which contents of the cache gets evicted if it fills up
  - Mapping scheme determines where to put the entries (such as direct-mapping)

Cache misses are costly

- Miss penalties are usually much greater than cache hit times
- Writing our code carefully to improve temporal and spatial locality can help reduce cache misses


On to the main lesson: Matrix-Matrix Multiplication!