Hardware (addendum)

A computer is designed from the ground up to run programs and the CPU and memory are both necessary for a computer to do its work.

**Types of instructions:**

*Arithmetic/Logic*

*Memory*: Put data to/get data from memory

*Control Flow*

- branches
  - unconditional branches

Some special ones to switch to the OS, start other programs

Where does memory fit in?

There are only so many registers (we’d have to load all the data into them at some point anyway)

1. Every instruction has to be gotten from memory (see: Stored Program Concept) -- *Every instruction in a program has to be gotten from memory*
2. Any data has to be gotten from memory at some point
3. Data from hard drives (storage) is stored in memory

![Memory Hierarchy](image)

*Figure 2.1* The levels in a typical memory hierarchy in a server computer shown on top (a) and in a personal mobile device (PMD) on the bottom (b). As we move farther away from the processor, the memory in the level below becomes slower and larger. Note that the time units change by a factor of $10^5$—from picoseconds to milliseconds—and that the size units change by a factor of $10^{12}$—from bytes to terabytes. The PMD has a slower clock rate and smaller caches and main memory. A key difference is that servers and desktops use disk storage as the lowest level in the hierarchy while PMDs use Flash, which is built from EEPROM technology.

From:

Comparison of speeds:
Instruction Speed: 300ps
Caches Speeds (averaged): L1 1ns (3x) L1 6.5ns (19x) L3 15ns (45x) Memory 75ns (225x) and, at an extreme, Disk Storage 75ms (225,000,000x)

Hard Drive Access Times: (Hard Disk Drive, HDD) from
  Seek Time - Time to move head to track with arm
  Latency- Time for data to move under head
  Transfer rate (vs. Random access rate)- The rate data can be moved to memory (often reasonably fast once the right location is found.

SSD: see table here:

Control Unit:
Branch Prediction

Pipelining