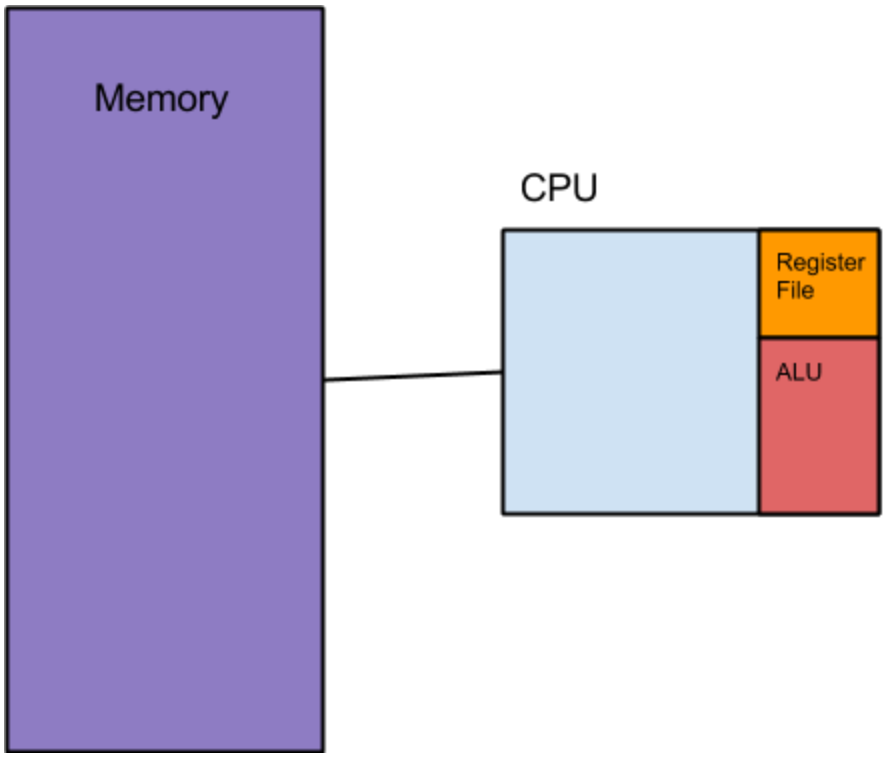
Hierarchy:

1. User
2. Programs
3. Operating System/Kernel, Compilers
4. Architecture, Computer Organization
5. Logic Gates
6. Physics, Quantum Mechanics, Transistors, CMOS, MOSFETS

- Input → (BLACK BOX) → output



- 8 bits = 1 byte = 8 binary digits (base 2)

reg 0	contains instructions
reg 1	
reg 2	
reg 3	
reg 4	
reg 5	
reg 6	
reg 7	
reg 8	
reg 9	
reg 10	

- instruction set register
- NZP sets negative, zero, positive for branch statements
- R0 = R1+R2
- Goes from memory, goes to register file, goes to ALU for computing
- Computer engineering tradeoff constraints
    - Speed/performance (MAX)
    - Heat/power (MIN)
    - Space/Area/Cost (MIN)
- Code

    for, if, def foo(), int R1 (pointer to an integer)

- Pointer

    (In 50) R0 = pointer to array = 0

    (In 51) R1 = cumulative sum variable = 0

    //length of array

    (In 52) R2 = 10

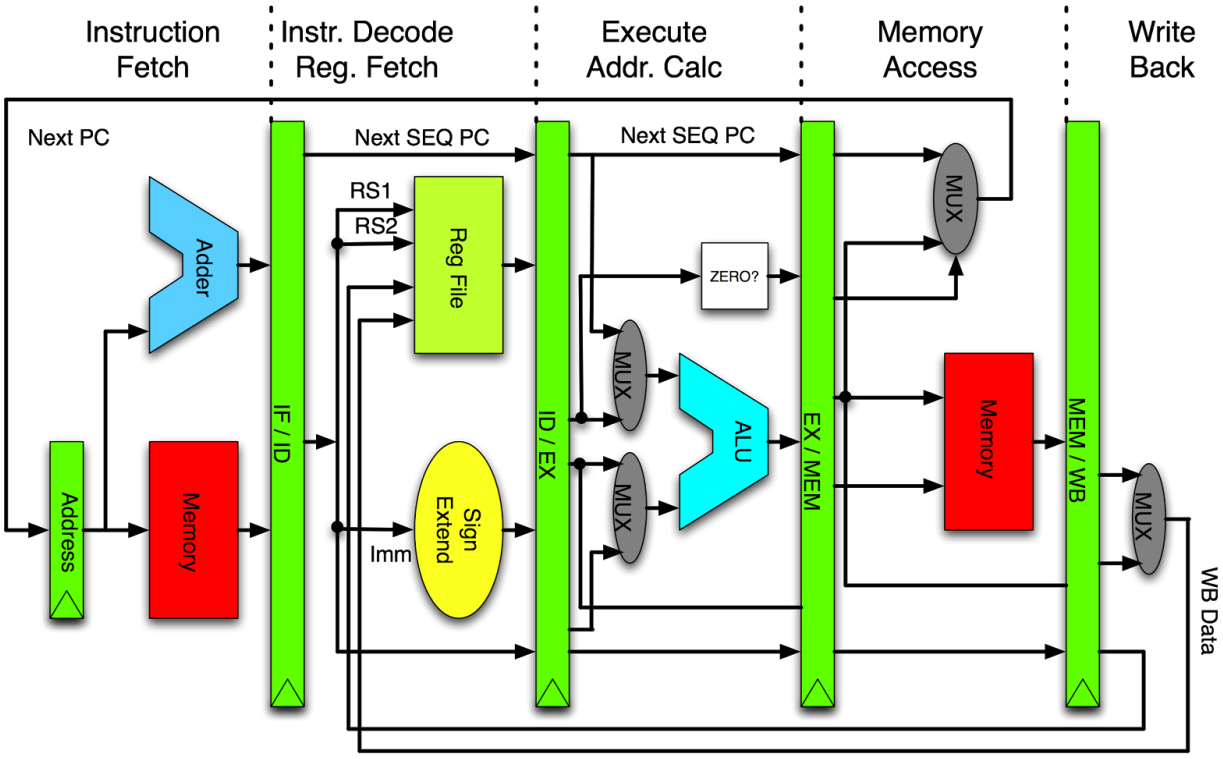
(In 53) loop: R1 = R1 + Mem[R0]

    (In 54) R0 = R0+1

    (In 55) R2 = R2-1

    (In 56) branch to 53 if not zero (Z)

- Data Hazard
    - RAW (Read After Write)
    - Read After Write dependencies stall and then data forward
    - WAR (Write After Read)
    - WAW (Write after Write)
- Steps
    - Instruction Fetch
    - Instruction Decode, Register Fetch
    - Execute, Addr. Calc
    - Memory Access
    - Write Back



- Out-of-order processing
- Hyperthreading (Symmetric Multi Threading)
    - Parallelism
- Cache
    - Temporary storage
    - L1, L2, L3