

Definition

- set of all instructions on a given computer architecture
- represented as
 - machine language
 - A sequence of zeros and ones, e.g. 0x83200002 → this is the sequence of zeros and ones the processor takes into its instruction register for decoding and execution
 - * Length varies can be many bytes long (up to 15 bytes on x86 CPUs)
 - assembly language
 - * somewhat human readable instruction representation
- interface between hardware and software
- ISAs from different vendors
 - different pros and cons
 - vary in terms of number of instructions
 - Intel x86, AMD64, ARM, MIPS, PowerPC, SPARC, AVR, RISC-V, ...
- ISA types
 - Complex Instruction Set Computer (CISC)**
 - Not only load and store operations perform memory accesses, but also other instructions
 - Design philosophy: many instructions, few instructions also for complex operations
 - Hundreds of instructions that include instructions performing complex operations like entire encryptions
 - Examples: x86 and x64 families
 - Reduced Instruction Set Computer (RISC)**
 - RISC architectures are **load/store architectures**: only dedicated load and store instructions read/write from/to memory
 - Design philosophy: fewer instructions, lower complexity, high execution speed.
 - Instruction set including just basic operations
 - Examples: ARM, RISC-V
 - One Instruction Set Computer (OISC)**
 - Computers with a single instruction (academic), e.g. SUBLEQ
 - * for academic purposes only
- open vs closed instruction sets
 - most ISAs covered by patents

Instruction [[Memory]]

- conditional branches
 - BNE (Branch if not equal)
 - BLT (Branch if less than)
 - BGE (Branch if greater of equal)
 - BLTU (Branch if less than unsigned)
 - BGEU (Branch if greater of equal unsigned)