programming language → restricted syntax → Formal language.

→ set of steps → algorithm → solve problems.

Abstraction → High-level language (closest to the problem domain)

↓

Compiler.

↓

↓

→ Assembly language.

↓ fast rep. of instructions.

→ easier to read for writing machine code.

Assembler

↓

→ Machine code. (object code / binary).

→ error-prone

↓ efficiency → save time,
productivity easy to read.
portability

Hardware / Software Stack in Computer.

Software

Application software

↓ CS2106

Operating System
- loader — load and run program.
- scheduler — balance computer's time between the softwares.
- device drivers.
- Compiler — CS4212
- Assembler
- Linker → link high-level code to various libraries to do various things.

Instruction Set Architecture (ISA) — specific rule to string together 0s and 1s

Hardware.
- Processor — computation / move data.
- Memory — store instructions req. by programs.
- I/O system — read from input / write to screen

↓

Datapath & Control design. — Control unit coordinate various paths for executing programs.

Collection of computational components that are linked together to execute instructions for program to run.

↓

CS2100

Digital Logic Design.

⇓

Circuit Design.

⇓

Transistors.

L3 cache → special, high speed memory → faster operations.

Core → Control Unit. → Coordinate components

CPU ↘

Arithmetic and Logic Unit.

↳ Collections of components.

↳ add numbers together.

↳ move data

↳ compare data.

↳ branches.

Computer Organization.

↳ how it works

↳ how everything is structured together and how they interact.

↳ how, based on these concepts, implement a computer system.

↳ Digital Logic Design → ISA

Cache → ordering of loops ?