Overview of Programming Languages (I)

History of Programming Languages

- Earliest digital computers **without memory** (The imitation Game Clip “Turing’s Machine” 2014 [https://www.youtube.com/watch?v=nmXzPgVjxRw](https://www.youtube.com/watch?v=nmXzPgVjxRw))
  - adjust gears, connect cables and flip switches

- Later computers **with punch card reader** (Punch Card from a Fortran program [https://en.wikipedia.org/wiki/Computer_programming_in_the_punched_card_era](https://en.wikipedia.org/wiki/Computer_programming_in_the_punched_card_era))
  - punch card: card-stock piece of paper with holes punched in it
  - data and instructions are holes
  - card reader reads holes
  - programmers **had to know the machine code** (different machines have different instructions)
  - **not easy to read and write**

- **Assembly language**
  - machine-dependent, CISC, RISC, close related to the machine architecture
  - the first abstraction of machine code, using text mnemonics to represent binary instructions and symbols to represent binary sequence
  - punch **card reader became interpreters**
  - more readable than pure machine code
  - **inefficient and error-prone** (count lines, no efficient flow control but JMP to lines)

---

**Machine Code**

**Assembly Language**    **Higher-order Languages**

1950s

Programming style is independent of any particular machine architecture
- After 1950s
  - PL were intended to bridge the gap between natural language and the machine instructions
  - Higher-ordered languages (our focus in CS333)
    - Independent of any particular machine architecture
    - Closer to natural language
    - Compilers/interpreters translate the programs into assembly languages/machine code