(Programming) Languages

- Formal Languages. Syntax and Semantics.
  - Wikipedia gives a silly definition using strings. Refers to Frege's Begriffsschrift as an example, but that language is 2-dimensional!
  - Syntax: Markdown, HTML, official Java grammar
  - Semantics: operational (how to run), denotational (what it means, using math)

(Programming) Languages

- Formal Languages. Syntax and Semantics.
  - Wikipedia gives a silly definition using strings. Refers to Frege's Begriffsschrift as an example, but that language is 2-dimensional!
  - Syntax: Markdown, HTML, official Java grammar
  - Semantics: operational (how to run), denotational (what it means, using math)
- Programming Languages
  - There are a lot of them. Some are truly bizarre. See Piet.
  - APL was weird, but almost mainstream. Too dense.
  - Genealogy and Influence (and more Influence).

(Programming) Languages

- Formal Languages. Syntax and Semantics.
  - Wikipedia gives a silly definition using strings. Refers to Frege's Begriffsschrift as an example, but that language is 2-dimensional!
  - Syntax: Markdown, HTML, official Java grammar
  - Semantics: operational (how to run), denotational (what it means, using math)
- Programming Languages
  - There are a lot of them. Some are truly bizarre. See Piet.
  - APL was weird, but almost mainstream. Too dense.
  - Genealogy and Influence (and more Influence).
  - Why Java?
    - ★ Popularity?
    - ★ Ecosystem?
    - ★ JCP?
    - ★ It is statically typed as well as decently designed?

(Programming) Languages

- Formal Languages. Syntax and Semantics.
  - Wikipedia gives a silly definition using strings. Refers to Frege's Begriffsschrift as an example, but that language is 2-dimensional!
  - Syntax: Markdown, HTML, official Java grammar
  - Semantics: operational (how to run), denotational (what it means, using math)
- Programming Languages
  - There are a lot of them. Some are truly bizarre. See Piet.
  - APL was weird, but almost mainstream. Too dense.
  - Genealogy and Influence (and more Influence).
  - Why Java?
    - ★ Popularity?
    - ★ Ecosystem?
    - ★ JCP?
    - ★ It is statically typed as well as decently designed?
    - ★ All, of course!

What is a Programming Language?

- Has a grammar which can be checked
- Is executable
- Is a medium of communication:

$$\text{Human} \Longleftrightarrow \textbf{Program} \Longleftrightarrow \texttt{Machine}$$

On Programs

- From 1 line to 300 million – Debian Lenny

On Programs

- From 1 line to 300 million – Debian Lenny
- On the other hand: integrated circuits with > 1 billion transistors. Which work.

On Programs

- From 1 line to 300 million – Debian Lenny
- On the other hand: integrated circuits with > 1 billion transistors. Which work.
- $\Longrightarrow$ Need principles.

Course objectives

1. Learn Java
2. Understand principles (by comparing with C and Caml and many others)
3. Start to learn the semantics of languages
4. Learn some principles of design and proper programming
5. Learn basic algorithms on lists, trees, and arrays.

Fundamentals of Imperative PLs

Fundamental constructions:

1. Assignment (syntax)
2. Declarations
3. Sequencing (Syntax: `;` or newline)
4. Test (Syntax 1, Syntax 2)
5. Looping (Syntax)