

# 1 Historical Perspective

Monday, August 24, 2026 11:24 AM

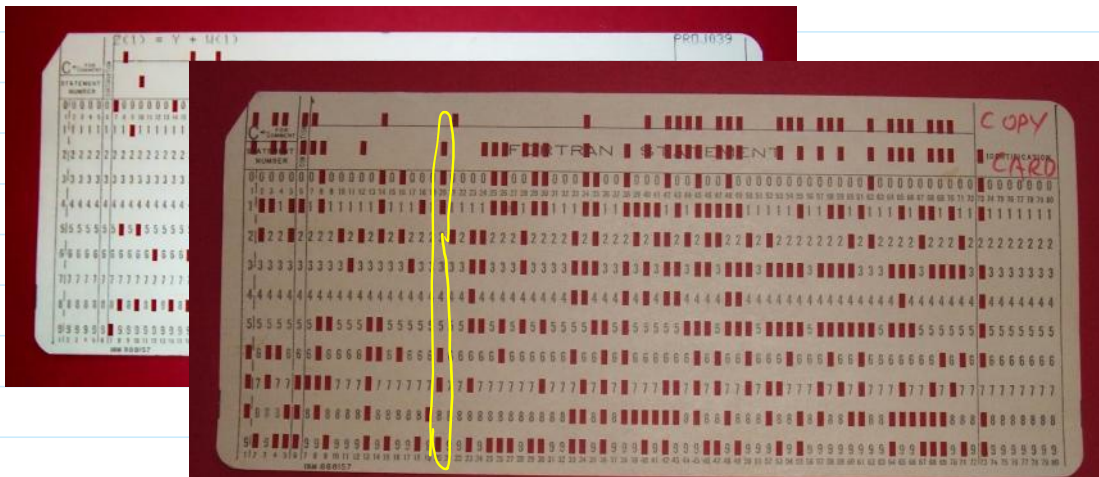
- Why Programming Languages?

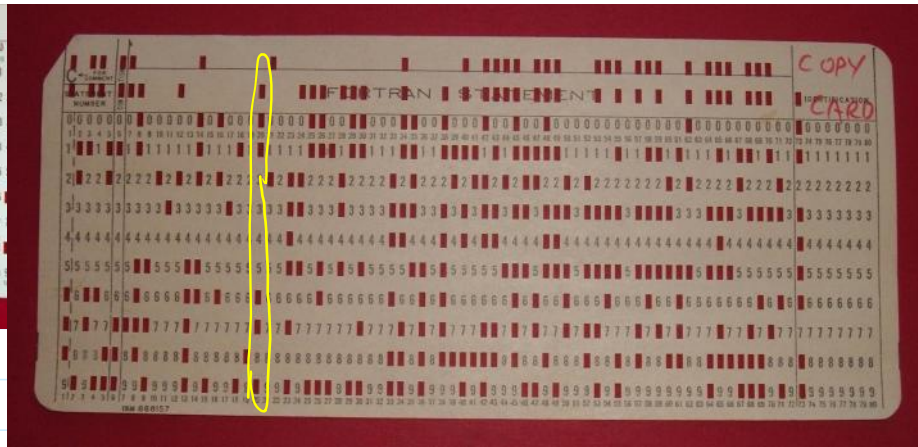
"in the beginning, Machines were Big, Problems small"



- Binary Code.

```
01010100 01101000 01101001 01110011 — Add 2 x
00100000 01101001 01110011 00100000
01110100 01101000 01100101 00100000
01110100 01110101 01110100 01101111
01110010 01101001 01100001 01101100
00100000 01110100 01101111 00100000 — sub x y
01101100 01100101 01100001 01110010
01101110 00100000 01100010 01101001
01101110 01100001 01110010 01111001
00101110 00100000 01001001 00100000
01101000 01101111 01110000 01100101 — store z in address 0x01100101
00100000 01111001 01101111 01110101
00100000 01100101 01101110 01101010
01101111 01111001 00100000 01101001
01110100 00100001
```





ComputerHistory.com

- Programming Languages
- '53 John Backus at IBM
  - "Make programmes more productive"
  - "Eliminate Bugs"
  - "It will take six months"
- '59 **FORTRAN** Formula Translator.
- '60 **Algol-60** First language with precise formal specification.
  - i Recursion!
  - Backus-Naur-Form
- **COBOL**

s?

- COBOL
- PL/1
- Algol-68
  - Templates
  - List Comprehensions
  - Exception handling
  - Fancy loop iterations.

s?

## • LISP

by John McCarthy

## • '70

- BCPL
- B
- C by Kernighan & Ritchie

UNIX

first personal computers

- BASIC
- LOGO for kids.
- Pascal by N. Wirth.

- Simula '79
  - Discrete event simulation.



- Prolog Colmerauer & Russell
  - Mathematical Logic.

## • '80

- Object Oriented Programming
- Smalltalk by Alan Kay

- Pascal → Object Pascal → Delphi

- C →
  - Objective C → MacOS
  - C with objects
  - C++ by Bjarne Stroustrup



## • Eiffel

- Perl by Larry Wall

## '90

- PCs are everywhere:
  - Visual BASIC

- '96 "the internet"

- HTML / CSS
- Java "Virtual Machine" → Android.
- JavaScript by Brendan Eich → Web App / Desktop.
- PHP

• 2000Q

• Python

Note: Languages don't die.

• 2010Q

Improve over older Languages:

- A better Objective-C
  - Swift
- A better Java
  - Scala
  - Kotlin
  - J++ → C# by Anders Hejlsberg
- A better C++
  - D Alexandrescu
  - Rust
  - Go
  - Zig
- A better JavaScript.
  - Dart
  - TypeScript
  - WebAssembly

• 2020's

• "the LLM is the new compiler"

Compiler : Source → binary known & deterministic.

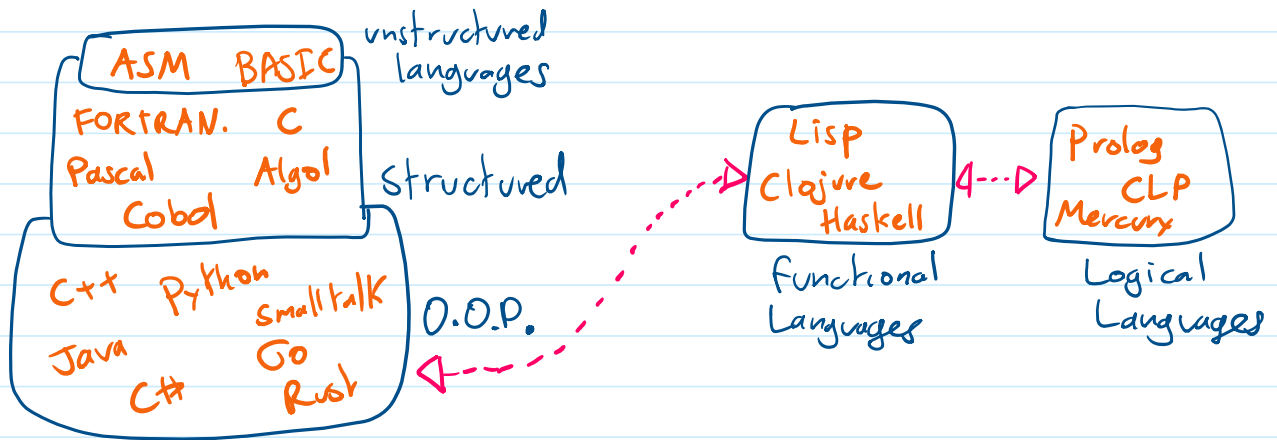
LLM : intent prompt → source. unknown & non-deterministic.



"Abstraction" - Create Software Modules that have goal Abstraction.

"Abstraction" - Create Software Modules that have goal Abstraction.  
"leaky Abstraction"

- Programming Language Taxonomy:



C++ '98 '03....'11, '17 '20 '23 '26 '29 '32...

—o—o—