The Beginnings - 1979

**FORTRAN  Lisp  Algol60  Pascal**

**B  BASIC  COBOL61  C**

- Beginnings of *high-level* programming languages.
- **Scientific and engineering** applications for practical use against assembly
- **Procedural** approach instead of focusing on future abstract object-oriented concepts.
1980 - 1994

C++ Ada MATLAB Objective-C
Perl Haskell Python VisualBasic

- Languages were becoming more object-oriented in the 1980s and 1990s
- (C++ / Python) result of personal projects of their creators as successors to other languages
- Python emphasizes code readability and OOP
- They became more and more high-level and useful.
1995 - 2009

Ruby  Java  Javascript  C#
OCaml  Scratch  GDScript  UnrealScript

- Web languages came out during this time
- A lot of languages for game engines were created in this time frame too.
- Scratch: Got kids into programming
2010 – Present Day

Rust  Swift  Dart  Go

• Coding is starting to be more geared to mobile development
• There also seems to be a lower barrier for entry when learning new coding languages
• Programming languages are becoming more efficient and optimal.
• First-class support of concurrency (with Go) and other features that weren’t used before
• Multi-paradigm languages (OOP, Functional)
What Features Do You Want in a Language?