CS 321 Programming Languages

Introduction

Baris Aktemur

Özyeğin University

Last update made on Monday 25th September, 2017 at 16:19.

This course

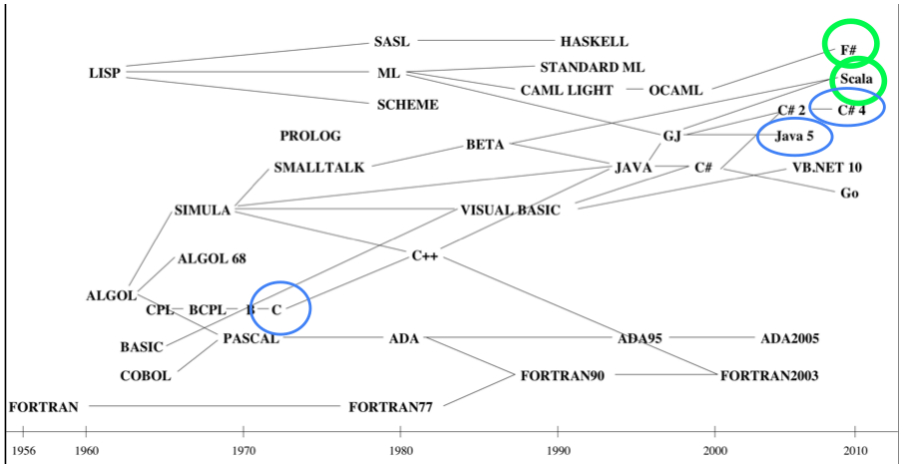
- Concepts of programming languages.
  - To help you have a deeper understanding of the execution model of programming languages.
  - To help you master your existing knowledge of programming.
  - To help you learn a new programming language easier.
- A new programming paradigm: Functional programming (using OCaml).

Why functional programming?

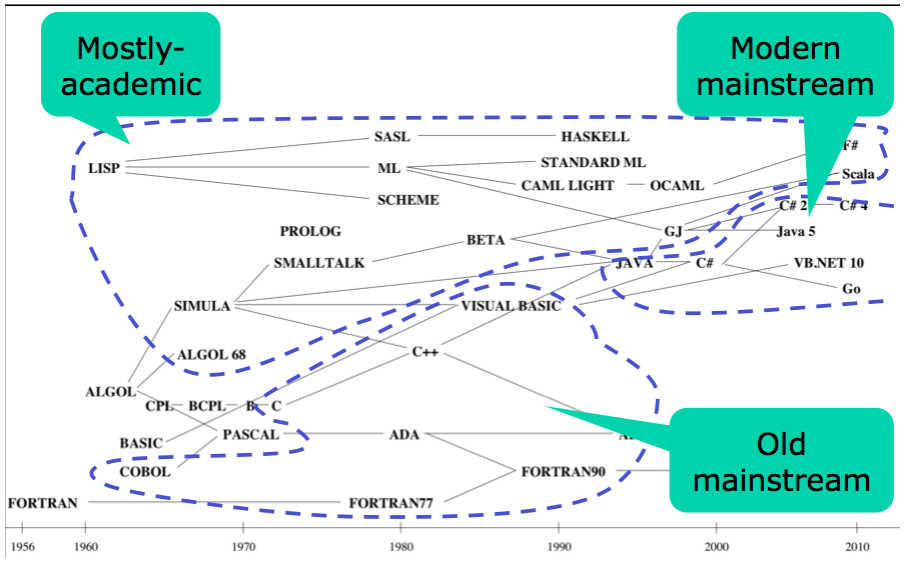
As stated in the preface of our textbook:

- The ML family of languages is ideal for implementing interpreters and compilers because of
  - datatypes, pattern matching, strongly typing.

  This leads to brevity and clarity.
- You shall be exposed to "different" views in programming to broaden your imagination. (Think outside of OOP)
- You'll see the shortcomings of mainstream programming languages (i.e. procedural and OO)
  - Generics (in ML since 1978), garbage collection (in Lisp since 1960)
- Functional and OO paradigms are merging.



[From PLC]



[From PLC]

Learning Outcomes

- Program in a functional language
- Implement the main building blocks of an interpreter
- Explain the principles of type inference and polymorphism
- Explain the operational semantics of an imperative language with pointer arithmetic
- Compare garbage collection algorithms

Syllabus

Everything is at

```
http://aktemur.github.io/cs321
```