

Course Information

2015/2016 1st Semester

CSIS0259 / COMP3259

Principles of Programming Languages

Basic Information

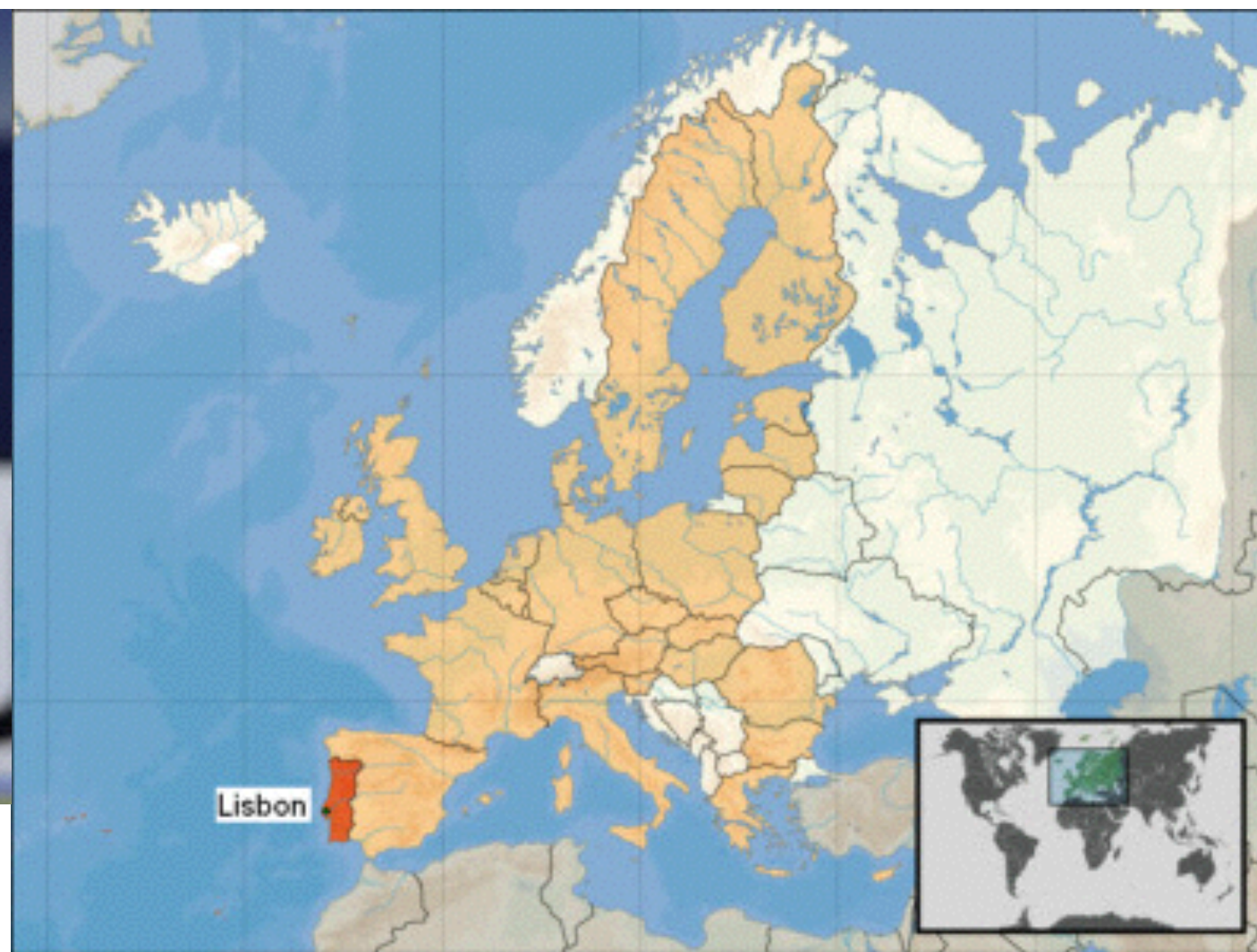
- Instructor
 - Dr. Bruno Oliveira (bruno@cs.hku.hk, CYC 420)
- Demonstrator
 - Xuan (Jeremy) Bi (xbi@cs.hku.hk), CYC 430)

About me

- Assistant Professor at HKU
 - Language: English (and Portuguese)
- Research Interests
 - Programming Languages
 - Functional Programming (especially Haskell and Scala)
 - Object-Oriented Programming
 - Modularity

More about me

- I come from Portugal
 - Best known these days as Cristiano Ronaldo's Land.
 - Macau was administered by Portugal until 1999.



What is this course about?

(Short version)

- How do Programming Languages work?

What is this course about?

- Learning the Principles of Programming Languages.
 - Basic concepts: Variable scoping, mutable state, closures, ...
 - Programming language semantics
 - Type systems
 - Programming paradigms (and how to program using different paradigms)

What is this course about?

- Programming Paradigms
 - Functional Programming:
 - First-Class Functions
 - No mutable state (Immutable variables)
 - Imperative Programming
 - Mutable State (Mutable variables)
 - Object-Oriented Programming
 - Objects, Inheritance

What is this course good for?

- Learn general concepts used by many languages
 - Java, C, Haskell, ... all have many concepts in common
 - By the end of the course you should be able to pick up new languages faster
- Learn to think differently about programming
 - Functional Programming vs Imperative Programming
 - Apply techniques from different paradigms to any languages
- Learn how to build your own languages
- To make you a better programmer!

Requirements

- This is not a beginners programming course
 - Students need to be comfortable programming in at least one Language (Java, C, C++, ...)
- Some basic knowledge about algorithms is desirable
- **Functional Programming** highly recommended!

Comparison to other Courses

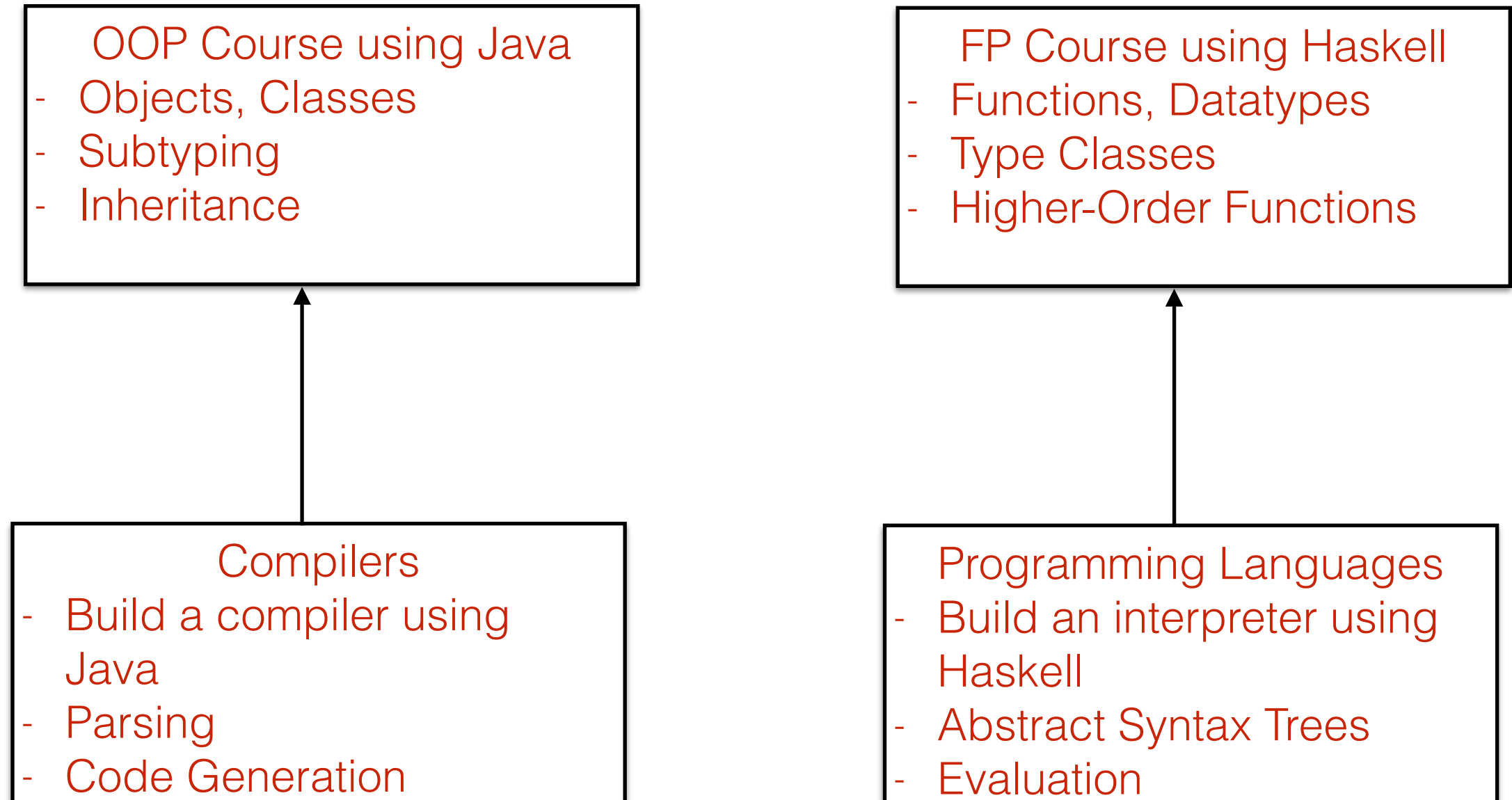
Functional Programming vs Programming Languages

- I teach 2 courses:
 - CSIS0259 / COMP3259: Principles of Programming Languages
 - CSIS2258 / COMP3258: Functional Programming
- What is the relation and differences between these 2 courses? Should I take both?

Functional Programming vs Programming Languages

- Functional Programming
 - **Goal: Learn how to do Functional Programming**
 - It will be a full course on Functional Programming and **Haskell**
- Programming Languages:
 - **Goal: Learn how Programming Languages work**
 - Develop interpreters for Programming Languages
 - **Haskell used to develop the interpreters**

Analogy



So...

- Object-Oriented Programming
 - Goal: Teach Object-Oriented Programming
- Functional Programming
 - Goal: Teach Functional Programming
- Programming Languages:
 - Goal: Teach how Programming Languages work and the mechanisms used in various paradigms
- Compilers:
 - Goal: Teach how to (efficiently) implement a Programming Language

What Course should I take?

- Functional Programming first
 - Works well
- Programming Languages first
 - Less recommended, **unless you are already a good programmer** and maybe you don't want to take Functional Programming.
 - First week of PL are spend on a crash course on Haskell (only features needed for implementing interpreters are needed)

Course Learning Outcomes

- Programming language fundamentals
 - Be able to understand the fundamental principles underlying various programming languages features
- Programming language design and implementation
 - Be able to design and implement simple programming languages
- Programming Paradigms
 - Be able to understand and apply ideas from different programming paradigms

Methodology of the course

- Learn by doing
 - You will implement programming languages with the concepts learned in class
- Growing a Language with features
 - Start with a small language (arithmetic expressions) and add more features (conditionals, function definitions, higher-order functions, mutable-state, ...)
- Use Haskell as Implementation Language

<http://www.haskell.org/haskellwiki/Haskell>

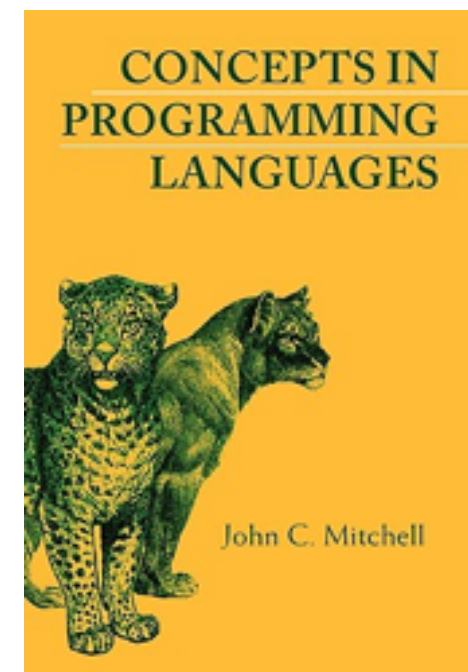
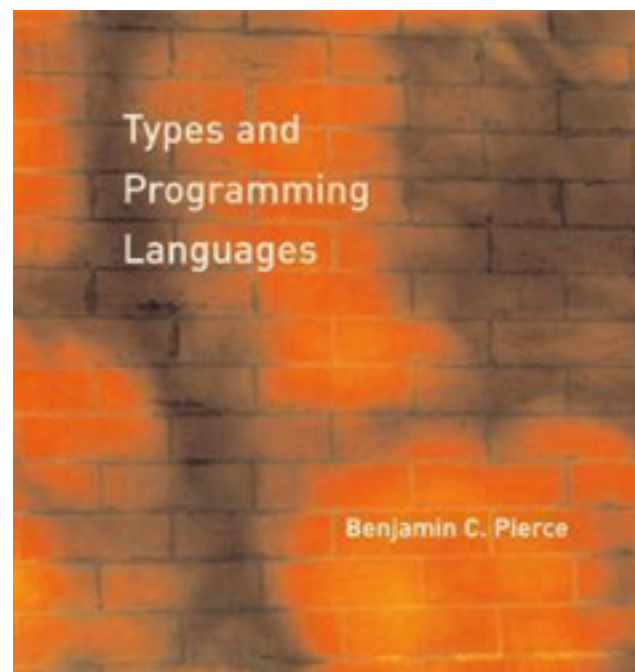
Schedule

Week	Topic
<i>1</i>	<i>Introduction/Haskell</i>
<i>2</i>	<i>Basic Expressions</i>
<i>3 - 4</i>	<i>Closures and first-class functions (Functional Programming)</i>
<i>5 - 6</i>	<i>Recursion and Data Abstraction</i>
<i>7 - 8</i>	<i>Errors and Mutable State (Imperative Programming)</i>
<i>9 - 10</i>	<i>Objects and Inheritance (Object-Oriented Programming)</i>
<i>11 - 12</i>	<i>Mini JavaScript</i>

Warning: Content may change according to progress!

Basic Information

- Reference Books and Materials
 - Anatomy of Programming Languages (book in development by Prof. William Cook, with my own contributions)
 - Types and Programming Languages (Benjamin Pierce, MIT Press)
 - Concepts in Programming Languages (John C. Mitchell, Cambridge Press)



Basic Information

- Reference Books and Materials
 - Coursera Programming Languages Course (By Dan Grossman, University of Washington)

<https://www.coursera.org/course/proglang>

```
fun append (xs, ys) =  
  if null xs  
  then ys  
  else (hd xs):: append (tl xs, ys)  
  
fun map (f, xs) =  
  case xs of  
    [] => []  
  | x :: xs' => (f x)::(map (f, xs'))  
  
val a = map (increment, [4,8,12,16])  
val b = map (hd, [[8,6],[7,5],[3,0,9]])
```

Lectures and Tutorials

- Monday
 - Time: 12:30 ~ 14:20
 - Venue: MB142
- Thursday
 - Time: 12:30 - 13:30
 - Venue: CPD-3.41

Tutorials

- Tutorial Participation (5% of total marks)
 - Please attend the tutorials!
 - Attendance will be recorded.
 - Please answer/raise questions during tutorial.

Consultation Hours

	Monday	Tuesday	Wednesday	Thursday	Friday
09:30~10:20					
10:30~11:20					
11:30~12:20					
12:30~13:20	Lecture			Lecture/Tutorial CPD-3.41	
13:30~14:20	MB142				
15:00~16:00				Consultation hour Jeremy (CB 430)	
15:30~16:20					
16:30~17:20					
17:00~18:00	Consultation hour Bruno (CB 420)				

If you want to meet outside consultation hours, please send me an email to arrange a meeting.

Policy

- Late assignments
 - upto 1 day late submissions (**15% of marks removed**)
 - upto 3 days (**30% of marks removed**)
 - more than 3 days (**not accepted**)
- Collaboration in study groups is encouraged, but you should **write your own program for the assignments**.
- Plagiarisms will be taken seriously! (Can be reported to the university).

Assessments

- Assignments (30%)
- Tutorial (5%)
- Mid-term examination and quizzes (15%)
- Final examination (50%)

Communication Channels

- Please come to us if you have any difficulties in the course
- There are several ways to contact and get in touch with us:
 - email
 - newsgroup
 - consultation hours
 - ...