



中国科学技术大学  
University of Science and Technology of China

# 编程语言及原理类 课程设置及教学内容探索

张昱

中国科学技术大学 计算机科学与技术学院

<http://staff.ustc.edu.cn/~yuzhang>

NASAC2018 系统软件教育论坛

2018年11月23日 深圳



# 报告目录

-  **1 问题和业界现状**
- 2 编程语言及原理类课程体系**
- 3 《程序设计语言基础》课的探索**
- 4 总结与期望**



# 讲授基础语言原理

Thomas Ball, Benjamin Zorn;

CACM2015.5

工业界期待迎来更多学习了程序设计语言原理的毕业生。

## 编

程序员日益成为紧缺人才，这一问题已得到广泛关注（请查看 [www.code.org](http://www.code.org) 网站上的相关内容，或观看 2013 年和 2014 年 12 月计算机科学周期间举办的“编码一小时” (Hour of Code) 课程）。尽管会编程在支持人们发挥创造力方面拥有巨大潜力，但我们应记住，编程语言仍将继续发展，以解决层出不穷的新问题，而语言原理总是万变不离其宗。如本观点文章所论述，语言基础在本行业所使用的复杂软件系统的设计与实现中发挥日益重要的作用。工业界亟需更多语言原理方面的人才，为客户提供可靠而高效的软件解决方案。

在过去，为应对复杂系统在设计<sup>1</sup>与实现方面的难题，很多重要语言原理应运而生。John McCarthy 在 1959 年前后为 Lisp 语言提出了垃圾回收机制，而该机制目前广泛应用于 Java 和 C# 等现代编程语言以及

语言基础在本行业所使用的复杂软件系统的设计与实现中发挥日益重要的作用。工业界亟需更多语言原理方面的人才，为客户提供可靠而高效的软件解决方案。

被广泛使用的编程语言奉为圭臬。<sup>2</sup> 类型系统按照程序表达式计算出的值来对其进行分类，<sup>3</sup> 让编译程序可以证明某类错误不存在，更高效地优化编码。Hoare 的断言方法为如何确定程序的正确性提供了框架。<sup>4</sup>

我们希望构建更加复杂的系统，要想以可预见的方式实现可靠的系统，能在更高层次表达编程人员意图的全新机制必不可少。随着新类型的系统不断问世，为了让更多人员能对此类系统编程，就必须设计全新的编程语言。表达编程人

企业和社会团体抽象的弱类型系统（此处为 C 语言）就会付出惨重代价。编程的可靠系统，互联网公司都已开发编程语言。例如，Go 语言 (<http://golang.org>)、Rust 语言 (<http://rust-lang.org>) 以及 Microsoft 的 Sing# 语言 (<http://singularity.codeplex.com/>)。这些语言通过可为程序执行提供更多安全保障的全新类型系统提高了编程水平。

领域专用语言 (DSL) 进一步提高了编程水平，通过限制表达能力实现了较通用语言更高层次的行为保证。SQL 数据库查询语言是一个经典例子，该语言基于关系代数，<sup>5</sup> 支持复杂的查询优化。

DSL 仍将在行业中派上用场。Google 的 Map/Reduce 数据并行执行模型<sup>6</sup> 推动了基于 SQL 的 DSL 的大量出现，其中包括 Yahoo 的 Pig



中国科学技术大学  
University of Science and Technology of China

# 高校如何应对业界变化？ 规模日趋庞大+多语言混合+异构





# 编程语言及原理类课程现状

- 偏重讲授多门编程语言
  - 如C/C++、Java、Python、Web编程、Android编程等
- 部分学校讲授编译原理和技术
  - 学时少，多数学校实验微乎其微
- 个别学校讲授不同编程范型等知识
  - 如北京大学裘宗燕教授讲授的《程序设计技术和方法》、我校的《程序设计语言基础》

缺乏统一管理  
部分内容重叠

照本宣科  
缺乏实践

师资、教材、范例缺乏  
学生知其然而不知所以然



# 报告目录

1

问题和业界现状

2

编程语言及原理类课程体系

3

《程序设计语言基础》课的探索

4

总结与期望



## 应对的基本思路（程序语言类）

- 第1门程序设计：精讲具体语言+实践
- 第2门程序设计：实践+设计方法学
- 编译原理和技术：基本原理+重实践
- 程序设计语言基础：不同语言特征及其语义和实现影响因素





# 我校程序语言系列课程简介

## □ L1 程序设计语言课程

- 程序设计I (C语言) 40+30学时, 2.5学分, 一上, **通修**
- 程序设计II 40+40, 3, 一下, **通修**
- Java /Android软件开发基础 2学分, **选修**

## □ L2 编译原理和技术

- 分层次 60+40学时, 4学分, 三上, **专业核心课**

## □ L3 程序设计语言基础 (2013年起开设, Stanford CS242: PLs)

- 60/20, 3.5学分, 三下, **专业方向课**  
(并行计算 2.5学分)



# 程序设计语言基础

## □ 介绍各种语言背后的通用概念和理论

- 概念上，涵盖**命令式语言**、**函数式语言**、**面向对象语言**、**并发和并行语言**以及**逻辑语言**的各种基本语言设施，包括控制结构、作用域、内存管理、高阶函数、继承、并发机制、新型并行编程模型等
- 理论上，介绍各种基础语言设施的**形式化语义**，包括静态语义和操作语义，以及相应的程序验证技术，如Hoare逻辑和类型系统等



# 课程体系的改进之处

## □ 程序设计I：精讲 + 实践

增加案例式教学，总结语言本质，养成查语言规范的习惯

## □ 程序设计II：上规模的实践 + 程序设计方法学

与数据结构同期或滞后开设，研制编程作业的代码框架

## □ 其他语言课：比较法讲授 + 上规模的实践

精心设计课程实践项目

## □ 编译课

补充现代编译技术，强化实践，案例设计、软件工程



# 课程体系的改进之处

## □ 程序设计语言基础课

引入有代表性的、具有不同编程范型和设计目标的编程语言及其编程实践项目来加深学生对课堂所学内容的领会

## □ 其他

- 一年级增加Python：易于上手，强大的软件包  
=> 解决现实世界问题，激发学习兴趣



# 与ACM/IEEE CS2013的对应

- ACM/IEEE CS2013: 18个知识领域(KA)
- 编程语言和原理类课程有关的KA
  - 程序设计语言PL、并行和分布式计算PD、软件开发基础SDF、系统基础SF
  - 知识点: 核心1级、核心2级和选修
  - 课程与PL、PD、SFD和SF中知识主体间的关系(下页)

	核心 1 级课时	核心 2 级课时	是否 选修	涉及的课程
PL/面向对象程序设计	4	6	否	程序设计语言基础
PL/函数式程序设计	3	4	否	程序设计语言基础
PL/事件驱动和反应性程序设计		2	否	程序设计语言基础
PL/基本类型系统、PL/程序表示 SDF/程序设计基本概念	1+0 10	4+1	否	程序设计 I
PL/语言翻译与执行		3	否	编译原理和技术
PL/语法分析、PL/编译语义分析 PL/代码生成、PL/运行时系统 PL/静态分析、PL/高级程序构造			是	编译原理和技术
PL/类型系统、PL/形式语义、 PD/形式模型和语义学			是	编译原理和技术、程序设计语言基础
SF/并行性	3		否	程序设计语言基础、并行计算
PL/并发与并行			是	程序设计语言基础、并行计算
PD/并行基础、PD/通信和协同	2+1	0+3	否	操作系统、程序设计语言基础、并行计算
PL/语言语用学、PL/逻辑式程序设计			是	程序设计语言基础
PD/并行算法、分析和编程		3	是	程序设计语言基础、并行计算
SF/状态与状态机	6		否	编译原理和技术



# 报告目录

1

问题和业界现状

2

编程语言及原理类课程体系

3

《程序设计语言基础》课的探索

4

总结与期望



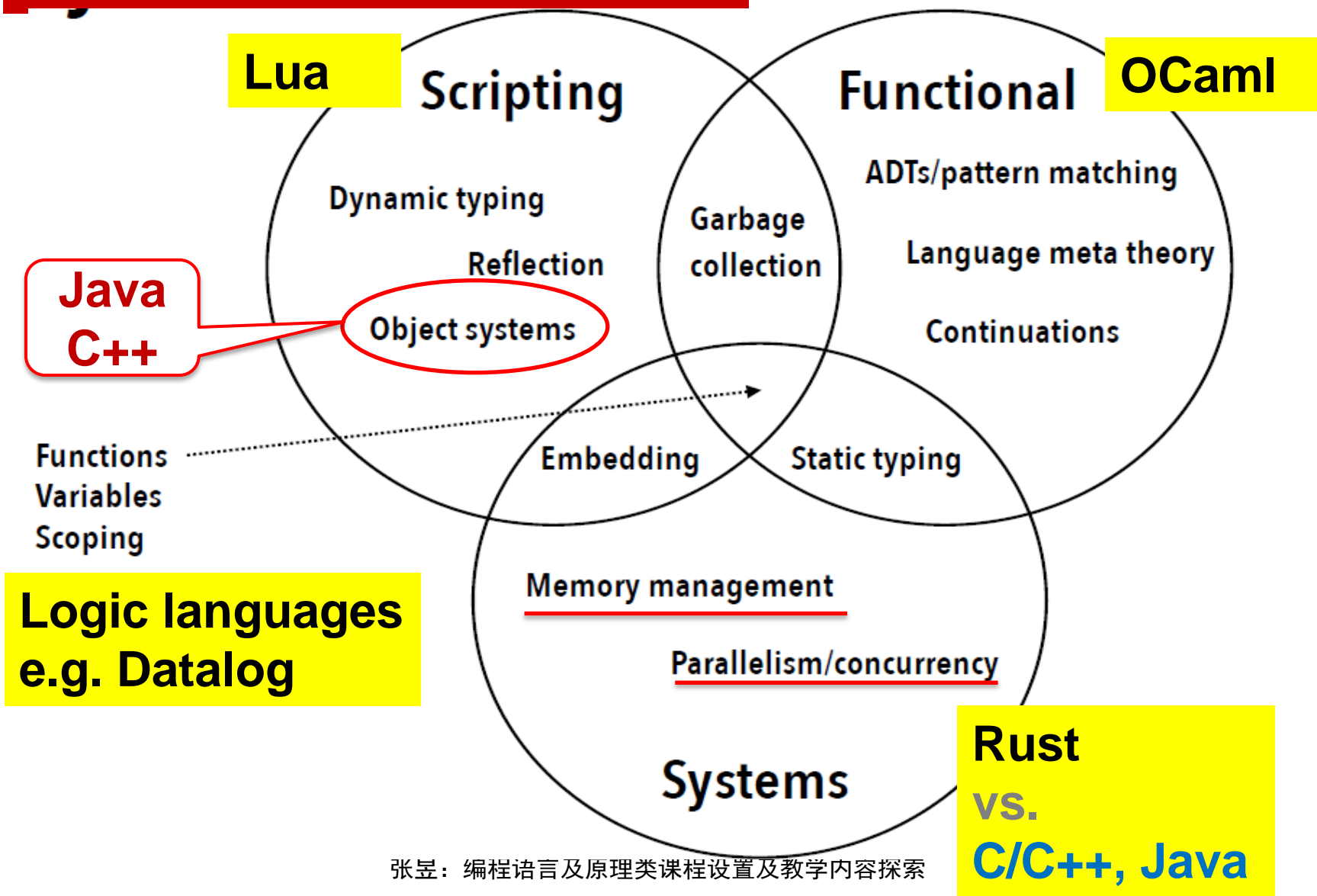
# 课程教学的主要思路

- 做中学
  - 使用和实现多种程序语言特征
- 理解现代编程语言的底层概念
  - 关注语义、关注编程模型、关注重要的语言特性
- 在语言设计决策中的一些权衡
  - 引入不同编程范型的多种编程语言
  - 同一语言特征在不同编程语言中的语义和实现差异，如对象系统、继承
- 学习设计自己的语言





# 教学内容





# 介绍的程序设计语言概念

- 作用域：静态 vs. 动态，参数传递，闭包
  - 对象系统：重载、继承、子类型等
  - 嵌入式语言
  - Lambda演算和类型系统
  - 控制流（协程、异常）
  - 类型系统与逻辑系统的对应
  - 内存管理
  - 并发、内存模型
- Lua**
- OCaml**
- Lua, OCaml**
- Datalog**
- Rust vs. C/C++, Java**



# 编程作业

作用域

**PA1: RPC Library - Serialization**

对象系统

嵌入式语言

**PA2: Roguelike**

Lambda演算和类型系统

**PA3: Type System**

控制流（协程、异常）

类型系统与逻辑系统的对应

**PA4: Logic Engine**

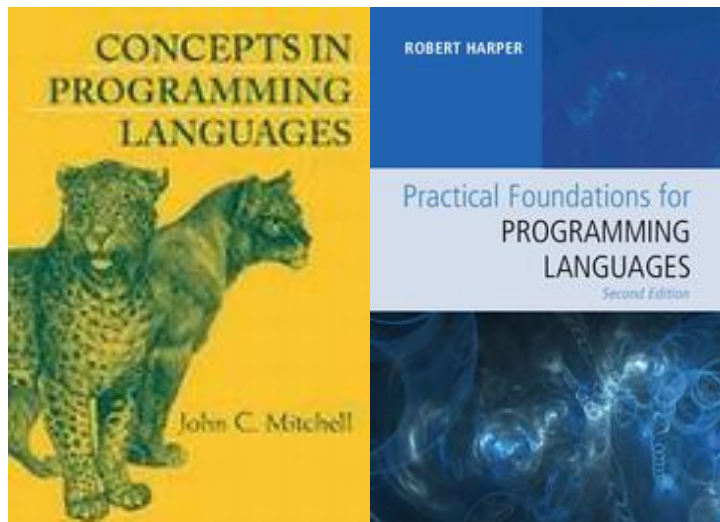
内存管理

并发、内存模型



# 教学相关的材料

- [CS242 at Stanford](#), [CIS500 at UPenn](#), [Cornell CS 3110](#)
- [Concepts in Programming Languages](#) by John C. Mitchell.
- [Practical Foundations for Programming Languages \(Second Edition\)](#) by Robert Harper. ([pdf](#)) Cambridge U. Press, 2016.
- [Types and Programming Languages](#) by Benjamin C. Pierce. MIT press, 2002. [[pdf](#)]
- [Programming Language Foundations](#) by B.C. Pierce.



More  
from Web



# 使用的工具



**REPL(Read-Eval-Print-Loop), Terminal multiplexer: tmux**

plazza

01116701 ▾

Q & A

Resources

Statistics

Manage Class

University of Science and Technology of China - Spring 2018

**01116701: Foundations of Programming Languages**



# FOPL 2018

- 25名学生选课：有挑战的课程
- 1名TA
- 课程网站
  - <http://staff.ustc.edu.cn/~yuzhang/fopl/>
  - <https://github.com/ustc-fopl/2018s>
  - <https://github.com/ustc-fopl/2018s-final-projects>



Group	paper/topic
1	xWIDL: Modular and Deep JavaScript API Misuses Checking Based on eXtended WebIDL
2	Dynamic Race Detection for C++11
3	Cambricon: An Instruction Set Architecture for Neural Networks
4	<del>TensorFlow: A System for Large-Scale Machine Learning</del>
5	Topic:Prolog
6	Repairing Sequential Consistency in C/C++11 [PLDI 2017]
7	Design and implementation of lua
8	The Java memory model
9	Foundations of the C++ concurrency memory model
10	FastTrack: Efficient and Precise Dynamic Race Detection
11	Partial Redundancy Elimination
12	Into the depths of C: elaborating the de facto standards [PLDI 2016]
13	DataCollider: Effective Data-Race Detection for the Kernel
14	Compiling without Continuations
15	The abstraction of STM in Haskell
16	TensorFlow: A System for Large-Scale Machine Learning
17	Self ideas
18	TVM:An Automated End-to-End Optimizing Compiler for Deep Learning





# 实践小结

## □ “好”的方面

- 提供代码例子：便于学生快速结合例子学习新语言
- 课程实践有代码框架和文档

The screenshot shows a Piazza class page for course 01116701. The page is titled "Class at a Glance" and is updated 3 minutes ago. It displays several key statistics:

- no unread posts (81 total posts)
- no unanswered questions (276 total contributions)
- no unresolved followups (46 instructors' responses)

Additional statistics include 41 students' responses and a 66 min average response time. Below the statistics, it shows "Student Enrollment" with 26 enrolled students. On the left side, there is a list of posts, including one titled "请大家填写校对分组情况到 go..." and another titled "请问对Final Project的Presentation的打...".





# 实践小结

## □ “欠缺”之处

- 学生主动交流不够
- 作业延期策略难以执行
- 需要布置小作业督促学生课后阅读和实践
- 没有合适的教材供学生系统学习
- 学时与实际任务的权衡



# 报告目录

1

问题和业界现状

2

编程语言及原理类课程体系

3

《程序设计语言基础》课的探索

4

总结与期望





# 总结与期望

- 业界需求=> 加强程序语言基础的能力培养
- 程序设计语言类课程体系有待重新规划
- 语言基础方面的教材极其缺乏
- 课程实践方案的设计非常重要
  - 深度 和工作量的权衡
- 师资
- .....



中国科学技术大学  
University of Science and Technology of China



谢谢