

《计算机程序的构造和解释》

Lab 09: Interpreter, Tail Recursion, Macros

助教：李晨曦、李煦阳、吴羽、徐鼎坤、张天昀

2022 年 12 月 07 日

本节实验课内容

- ▶ 18:30 - 19:00 讲讲 hw08
- ▶ 19:00 - 19:30 讲讲编程语言范式
- ▶ 19:30 - 20:20 完成 lab09、答疑

上周作业讲解

1. 提问的基本都是自己的代码都解释不清楚的
2. 有请 xygg

编程语言范式

我们这门课会学哪些编程语言？

- ▶ 命令式：Python
- ▶ 函数式：Scheme
- ▶ 声明式：SQL（结构化查询语言）

他们各自有什么特性？有什么区别和优劣？

命令式 (Imperative)

关注程序执行的步骤，告诉计算机每一步该做什么

例如：从南京大学仙林校区开车去鼓楼校区

1. 开车：插钥匙，踩刹车，点火，松手刹，挂挡，踩油门……
2. 导航：走九乡河西路向北，走 G312 到玄武湖立交……

优缺点：

1. 符合计算机的冯诺依曼架构（状态机）
2. 新手容易看懂学会
3. 程序会变得逐渐复杂，抽象能力差

面向对象程序设计 (OOP)

将数据和方法结合到一起，形成对象

例如：从南京大学仙林校区开车去鼓楼校区

1. 汽车提供“驾驶”方法，驾驶员无需关心车怎么开
2. 导航：走九乡河西路向北，走 G312 到玄武湖立交……

优缺点：

1. 降低程序的复杂性和耦合性，形成易于复用的模块
2. 不同的编程语言有不同的 OOP 设计（例：C++ v.s. Java）
3. 设计优劣非常依赖于程序员的经验与能力（例：“四人帮”）

函数式 (Functional)

关注做什么，不关注怎么做

- ▶ 函数是一等公民
- ▶ 数据不可变、函数无副作用
- ▶ 递归!

常见函数式编程语言：LISP、Haskell、Clojure……

例如： $f(x) = x^2 \implies (\text{define } (f\ x) (*\ x\ x))$

优缺点：

1. 有一种数学的美
2. 代码逻辑易于理解
3. 很难写，“纯”函数式太难用（例如：怎么开车？）

命令式 + 函数式

函数式逻辑清晰易于表达，为什么不拿来用呢？

- ▶ C++11: Lambda 函数、Functor
- ▶ Java¹: Lambda 函数、闭包、Stream API

例如：排序，`sort(begin, end, lambda)`

流式调用（Fluent Interface，也就是链式调用）：

```
Stream.of(1, 2, 3, 4, 5)
    .map((x) -> x * 2)
    .filter((x) -> x < 10)
    .forEach((x) -> System.out.println(x));
```

优缺点：

- ▶ 逻辑清晰，易于表达，就像数据在流动一样
- ▶ 难以调试

¹虽然 8 这个版本号很老，但是 Java8 是 2014 年发布的

声明式 (Declarative)

只告诉计算机你要做什么，剩下的事情由计算机来解决

例如：

- ▶ 从南京大学仙林校区开车去鼓楼校区
- ▶ 从图书馆找出所有作者包含 XXX 的书

优缺点：

- ▶ 逻辑非常清晰（废话）
- ▶ 运行效率依赖于程序语言的具体实现

C# 的 LINQ 语法：

```
List numbers = new List {1, 2, 3, 4, 5};  
List oddNumbers = (SELECT x IF x % 2 == 0  
                    FROM numbers).ToList();
```

编程语言范式

我们这门课会学哪些编程语言？

- ▶ 命令式：Python
- ▶ 函数式：Scheme
- ▶ 声明式：SQL（结构化查询语言）

他们各自有什么特性？有什么区别和优劣？

- ▶ 程序设计语言并没有优劣之分
- ▶ 应该根据实际需求选择最合适的编程范式、语言和工具

作业与课程安排

本周的作业

- ▶ 12 月 10 日截止: Lab09
- ▶ 12 月 13 日截止: Homework09
- ▶ 12 月 25 日截止: Project04