



## 关于函数式编程的一点杂谈



Felis sap... 

函数式编程、编程语言、编程 话题的优秀回答者

已关注

rainoftime、开源哥、圆角骑士魔理沙、祖与占、考古学家千里冰封等 147 人赞了该文章

inspired by [前端开发js函数式编程真实用途体现在哪里？](#) - 知乎

- Y 的确是一种在 call by name 规约下，无类型 lambda calculus 使用自己的 term 表达的不动点组合子（以下称为 fix）。其在 call by value 规约下的对应版本是 Plotkin 的 Z（对 Y 进行 eta 展开的产物）。JavaScript 是 call by value 的语言。
- 在各种带类型 lambda calculus 中，fix 的类型可以表达，但其类型是 uninhabited 的，不能用自己的 term 写出来。可以通过将 fix 直接作为一个新的 primitive 并给定对应的 typing/reduction rule，或者加入 recursive type，这样可以重新实现 fix，通过 fix 实现递归函数，不过整个系统会失去 strongly normalizing 的特性，Curry-Howard 对应的逻辑变得 unsound。
- 从实用的角度来看，没有人会直接将 Y 用于实现递归函数。编译器会直接将 fix 或者 letrec 作为 primitive 实现。使用手写的 fix 实现递归函数时，一般会用这个 fix 做一些更有趣的事情（比如实现 memoization）
- 所以对函数式编程而言，Y 既谈不上实现递归的通用手段，也不是那么实用的东西（虽然确实有理论价值/有趣）

一点技术硬伤也就罢了。关键是“识别一个人是不是函数式编程小白的最简单方法”这样的说法我觉得挺扎眼的。不同的“大佬”总是可以划出一些不同的线，仿佛学过什么东西以后就能高人一等，比如读过什么书/paper，学过什么语言或者概念。就函数式编程这个例子，能从故纸堆里寻出的高级玩意儿实在多了去了，Y 组合子算什么，我也可以说“会手写 Zygohistomorphic prepromorphism 或者 Profunctor Optics 才不是小白” .....这有意思么？

编辑于 2017-05-19



赞同 147   33 条评论   分享   收藏   ...

文章被以下专栏收录



不动点高校现充部  
一切与编程语言理论、函数式编程相关的杂谈。

已关注

推荐阅读



函数式编程：柯里化的变型应用

发表于前端码农的...



[杂谈] 谈谈读书 (续)

陈天   发表于迷思



支教杂谈 (一)

陈思   发表于10英尺上...

杂谈  
昨天  
党积  
为什  
的爷  
近党  
所以  
欺世

33 条评论   切换为时间排序

写下你的评论...

Belleve   1 年前  
干得漂亮！ 然而真正的槽点难道不是——他不知道 STLC 是什么吗？  
24

梨梨喵   1 年前  
Y combinator這東西真正談用處在於Lambda definability與Recursiveness等價上, 其他的像關於遞歸函數實現什麼的都是相對比較次要的. 要說對於FP確實不算什麼很關鍵的東西。  
11

以上为精选评论

naiveman 回复 Belleve   1 年前





jie cui

1 年前

封面好评。。。晓美焰

👍 3



后缀自动机·张 回复 naiveman

1 年前

给大爷跪了。。。萌并不会

👍 1     查看对话



nameoverflow 回复 后缀自动机·张

1 年前

qc大爷抽空撕烤一下不就会了

👍 1     查看对话



winter

1 年前

我无意冒犯牛逼闪闪的贵圈，也并非函数式编程专家，我真觉得入门勉强吧，至于yc，当然更不是什么高端的东西，上世纪50年代的旧货罢了。

在我看来，会写yc说明不了任何问题，不会写yc说明是个小白。所以呢，其实我写了个yc就是写了yc而已，说明不了我很懂FP，我也确实不是很懂FP——只够分辨个把骗子而已。

关于几个技术问题，回应一下：

1. 是yc还是zc这个，评论里也是有人提到的，这个我写的这段长得像yc，按你们分析的芯里其实是zc，这个我觉得有道理，我不纠结，反正都不牛逼。
2. 说了一堆有类型的会给fix带来很多麻烦的事，呃，虽然我不知道为什么要说这个，但是很好。
3. 没有人会直接将 Y 用于实现递归函数，这个我也觉得没什么问题，而且我也确实写了不带Y的版本。

额外一说，我觉得楼上某位小朋友，觉得懂缩写牛逼这件事，挺low的。

👍 4



ETIN

1 年前

你们都是大佬~我等只能膜拜！

👍 赞



Eternal Chaos

1 年前

手写CPS转换算不算？



我粗浅地以为Y最初搞出来只是为了能最小化lambda calculus，从而让证明变得简单。少一个primitive 就能少一大堆case analysis。只是这个证明工具看起来很特别，最后被人以讹传讹了...

👍 3



naiveman 回复 winter

1 年前

我本来就很low啊，tapl看了一点点就弃了。这就跟知道茴字有几种写法一样啊。。。本渣就一个失败人士而已。。。写出来只是方便不知道但想知道的人去看wiki而已。

👍 赞    💬 查看对话



祖与占 回复 naiveman

1 年前

没觉得你有发个链接就显得牛逼的意思...反而觉得这么judge人的没意思

👍 6    💬 查看对话



Yutong Zhang

1 年前

y在cbv下diverge吧？

👍 赞



李约瀚

1 年前

昨天相关内容之后，马上去补了一波 Y，然后总有一种上当受骗的感觉，，

👍 赞



neo lin

1 年前

还是看不懂怎么办？

👍 赞



Kakyoin Noriaki 回复 winter

1 年前

如果单纯说会用JS写ZC的话，我感觉就算水平达不到函数式小白的人也能做到.....

如果说会证的话，估计也不难。

问题是您确定您的FPL和别人的FPL的定义是一样的？很多人眼中的FPL不是这样。

所以严格来说，会写YC/ZC只能说是辨别Lisp程序猿是否是小白的方式.....吧。

👍 1    💬 查看对话



Kakyoin Noriaki

1 年前

另一个槽点在于，就算不是FP小白，YC半个小时不到就能全部搞懂吧.....这样不方便卡人啊。

👍 2



rainoftime 回复 Kakyoin Noriaki

1 年前



Kakyoin Noriaki [回复](#) rainoftime

1 年前

更大的槽点是，“某个JS圈的大佬公然说STLC和FP没关系不属于Lambda公理体系”  
以及，“某个JS圈的大佬公然说FP和PLT没半毛钱关系”

 2 [查看对话](#)

dram

1 年前

Haskell 里面写 y combinator 需要一个 negative isorecursive type, 而且还会炸编译器 (ghc)  
(经典 wontfix bug)

 4