函数式编程 编程语言理论 Idris ♪ 修改

关注者

被浏览

48

他们也关注了该问题

2.942

Idris 支持 General Recursion, 那它的 typechecking 会不会死循 环? ▶ 修改

众所周知在 CoC 的框架体系内,Type 和 Term 完全等同,因而 typechecking 的关键步骤—— Unification——需要真的去...显示全部 >

关注问题

▶ 写回答

+ 激请回答

● 添加评论 ▼ 分享 ★ 邀请回答

查看全部 3 个回答



Felis sapiens 🛟

函数式编程、编程语言、编程 话题的优秀回答者

考古学家千里冰封、HOOCCOOH、罗宸、开源哥等 18 人赞同了该回答

当然可以死循环。total functional programming 语言的 termination checker 都是基于非常 naive 的语法特征去匹配的 (Idris 的文档太缺了,参考 Agda wiki 上的说法:

wiki.portal.chalmers.se...) , 但并不是所有 total 的函数都满足 primitive/structural recursion 的语法特征。

所以无论 Agda 和 Idris 都提供了"大爷我觉得这函数 total,编译器给我闭嘴"的功能。Agda 有 个 TERMINATING pragma (见Termination Checking) , Idris 有个 assert_total 函数 (见 Theorem Proving - Idris 0.99 documentation)。你用这功能堵上编译器的嘴,写个 diverging 的函数, 让类型检查器空转是完全 ok 的。

至于 Haskell 嘛。。捂脸。。不上依赖类型,我也有一百种让 GHC 死循环或者 panic 的方法(笑 编辑于 2017-01-28

▲ 赞同 18 ▼

● 添加评论

7 分享

更多回答



dram ♥ Bad Apple!!

2 人赞同了该回答

有可选的 totality checker, 如果检查不过的话, 在类型检查的时候是不会规约的

non-total definitions aren' t reduced when type checking because they are not welldefined for all possible inputs.

展开阅读全文 ~

▲ 赞同 2

■ 3 条评论

7 分享

★ 收藏

● 感谢



baozii

有空来我家吃饭吧

1人赞同了该回答

01 类型检测不会死循环,这是在语言类型系统设计之初就设定好的,你要了解详具体情况可以看一 下直觉主义类型系统(觉得会死循环不妨举个例子)。但是类型推导有可能。具体到idris,当然也 不会陷入死循环, 但类型标注不明时它会提示让你明确标注类型



关于作者



Felis sapiens

☆ 函数式编程、编程语言、编程 话题的 优秀回答者

♣ 电影旅行敲代码、Antokha Yuuki、 暮无井见铃也关注了她

回答

文章

关注者

185

40 14,871

● 发私信

被收藏 1次

代码

1人关注

灯语 创建

相关问题

怎样评价 LambdaConf 提出的 "函数式 编程技能表"? 9个回答

在 Haskell 等语言中是否无法表示 Functor 等的公理? 5 个回答

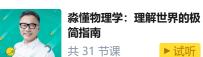


02 General Recursion 和你问的问题没关系呀,几乎所有流行的fp语言都是General Recursion,
类型检测的算法和性质与限定total的是一样的
展开阅读全文 ➤

▲ 赞同 1 ▼ ● 添加评论 ▼ 分享 ★ 收藏 ● 感谢 …

查看全部 3 个回答

什么是 Type Driven Development ? 个回答





相关推荐

数学妙啊! 妙! 张英锋 等 289,069 人读过





刘看山·知乎指南·知乎协议·隐私政策

应用·工作·申请开通知乎机构号

侵权举报·网上有害信息举报专区

违法和不良信息举报: 010-82716601

儿童色情信息举报专区

电信与服务业务经营许可证

网络文化经营许可证

联系我们 © 2018 知乎