

Types and Programming Language

枫聆

2021 年 3 月 29 日

目录

1 Introduction

2

Introduction

Definition 1.1. A **type system** is a tractable syntactic method for proving the absence of certain program behaviors by classifying phrases according to the kinds of value they compute.

type system 是一种用于证明某些确定的程序行为不会发生的方法，它怎么做呢？通过它们计算出值的类型来分类，有点抽象... 我想知道 the kinds of value they compute 是什么？如何分类？分类之后接下来该怎么做？

Annotation 1.2. Being static, type systems are necessarily also **conservative**: they can categorically prove the absence of some bad program behaviors, but they can't prove their presence.

Example 1.3.

```
1 if <complex test> then 5 else <type error>
```

上面这个 annotation 在说 type system 只能证明它看到的一些 bad program behavior 不会出现，但是它们可能会 reject 掉一些 runtime time 阶段运行良好的程序，例如在 runtime 阶段上面的 else 可能永远都不会进。即 type system 无法证明它是否真的存在。