

数学

✎ 修改

"1 + 1 = 2" 是定义, 还是定理? ✎ 修改

如果是定理, 怎么证明?

✎ 修改

- 关注问题
- ✎ 写回答
- + 邀请回答
- 💬 添加评论
- 🚩 分享
- ★ 邀请回答
- 🚩 举报
- ...

关注者

161


被浏览

53,549

他们也关注了该问题



查看全部 27 个回答

 **Felis sapiens** 🌟
函数式编程、编程语言、编程 话题的优秀回答者

头顶青天红美铃、开源哥、圆角骑士魔理沙、hhhhhhhhh、nameoverflow 等 25 人赞同了该回答

@Belleve 提到的在 CoC 里用 lambda encoding 来表示 inductive definition 的 paper:

cambridge.org/core/jour...

tftp2016.org/papers/TFP_...

我也来凑个热闹好了, 用 GHC 的类型系统来“证明” $1 + 1 = 2$:

```
{-# OPTIONS_GHC -Wall #-}
{-# LANGUAGE DataKinds #-}
{-# LANGUAGE GADTs #-}
{-# LANGUAGE ScopedTypeVariables #-}
{-# LANGUAGE TypeFamilies #-}
{-# LANGUAGE TypeOperators #-}

import Data.Type.Equality

data Nat
  = Zero
  | Succ Nat

type family Add x y where
  Add 'Zero n = n
  Add ('Succ m) n = 'Succ (Add m n)

type One = 'Succ 'Zero
type Two = 'Succ One

onePlusOneEqualsTwo :: Add One One ~: Two
onePlusOneEqualsTwo = Refl
```

用 Idris 或者 Agda 之类的其他 dependently-typed language 写会好看很多, 不过中心思想都一样——规定好什么是自然数 (Peano numbers)、什么是 1 和 2 (1 是 0 的后继, 2 是 1 的后继)、什么是自然数的加法、什么是等于 (propositional equality), 然后 type checker 计算 normal form 并进行匹配。

以上这个例子太 trivial 了, 来证个加法的结合律和交换律看看:

```
data SingNat :: Nat -> * where
  SZero :: SingNat 'Zero
  SSucc :: SingNat n -> SingNat ('Succ n)
```

腾讯云新注册用户域名抢购1元起 广告 X

.com

.cn

.xyz

腾讯云新用户域名抢购1元起

.xyz

更多多种爆款组合选购

➤

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

624

文章

40

关注者

14,871

已关注

发私信

被收藏 3 次

PL

Isdsjy 创建

2 人关注

Math

tomsu 创建

0 人关注

开发

Julia的骆驼 创建

0 人关注

相关问题

高数中有哪些有意思的定理? 4 个回答

```

assoc :: SingNat x
      -> SingNat y
      -> SingNat z
      -> Add x (Add y z) ~: Add (Add x y) z

assoc SZero _ _ = Refl
assoc (SSucc x') y z =
  case assoc x' y z of
    Refl -> Refl

commune :: SingNat x -> SingNat y -> Add x y ~: Add y x
commune m n =
  case (lemma0 m, lemma0 n) of
    (Refl, Refl) ->
      case m of
        SZero -> Refl
        SSucc m' ->
          case n of
            SZero -> Refl
            SSucc n' ->
              case (commune m' n', commune m' n, commune m n') of
                (Refl, Refl, Refl) -> Refl

where
  lemma0 :: SingNat n -> Add n 'Zero ~: n
  lemma0 SZero = Refl
  lemma0 (SSucc n') =
    case lemma0 n' of
      Refl -> Refl

```

编辑于 2017-05-08

赞同 25

2 条评论

分享

收藏

感谢

...

收起

更多回答



winter

知乎吃枣药丸

263 人赞同了该回答

首先科普一些常识，像是 $1+1=2$ 这样的东西，从现实中抽象出来，是最基础的，可能也是最早的结论，大部分数学分支的起源也都是从现实中抽象出来的。

但是实际上公理并不是先抽象出来的就是公理，也不是简单的就是公理，而是用最少的假设推导出其它结论，所需要的那些假设。

在现代数学里面，存在不同的公理体系，从每一个完整的公理体系出发，理论上都能推导出整个数学。

比方说A B C D E五个结论，可能A B C作为公理能推导出D E，A B E也能推导出C D。

公理体系最开始一般都建立在一些数学的分支上，早年的公理后来都逐渐被发现一些问题逐渐完善，像是欧几里得最开始搞的五公设，后来到希尔伯特填坑出来21公理体系。

后来还产生了研究不同公理体系之间映射关系的数学分支范畴论，反正我这智商是搞不太懂，就不误导大家了。

展开阅读全文

赞同 263

33 条评论

分享

收藏

感谢

...

那些巧妙而复杂的高数定理证明，大牛也都能记得住吗？ 31 个回答

定理证明？ 2 个回答

数学上有哪些令人惊诧的定理？ 8 个回答

请问一下数学上是否有什么定理已经被证明，但是还没找到相对应的例子（实例）？ 8 个回答

相关推荐



森懂物理学：理解世界的极简指南

共 31 节课

试听



线代选讲：相似对角型的价值

★★★★★ 151 人参与



费马大定理

西蒙·辛格

2,812 人读过

阅读

广告 X

腾讯云新注册用户域名抢购1元起

.com

.cn

.xyz

腾讯云新用户域名抢购1元起

.xyz

更有多种爆款组合选购

>

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





hhhhhhhhh
玩完了就跑



56 人赞同了该回答

陈宇的基本对了。。。

作为一个小学生居然不觉得这个是定义我真是吓哭了。。。

自然数是根据 peano axiom 定义的数系，这个公理再证明中基本不需要的部分，我就掠过了。

公理：存在元素 0，属于自然数。

公理：任意自然数的后继数为自然数

(以上为 peano 公理的部分)

定义：自然数 n 的后继数记作 $n++$

定义： $0++=1$ ， $1++=2$

好了，我们接着来定义自然数加法

对于任意自然数 n ， $n+0=n$ ，对于自然数 m ， $n+(m++)=(n+m)++$ 。

[展开阅读全文](#)

赞同 56



9 条评论

分享

收藏

感谢



[查看全部 27 个回答](#)

