

Seaman.h.zhang

博客园 :: 首页 :: 新随笔 :: 联系 :: 订阅 [XML](#) :: 管理 34 Posts :: 0 Stories :: 2 Comments :: 0 Trackbacks

公告

昵称: seaman.kingfall
园龄: 4年3个月
粉丝: 4
关注: 1
[+加关注](#)

搜索

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

我的标签

[练习题\(6\)](#)
[合一\(3\)](#)
[递归\(3\)](#)
[中断\(2\)](#)
[类型变量\(2\)](#)
[数字\(2\)](#)
[列表\(2\)](#)
[Haskell\(2\)](#)
[recursive\(2\)](#)
[比较\(2\)](#)
[更多](#)

随笔分类

[Haskell\(2\)](#)
[Prolog\(32\)](#)

随笔档案

[2015年8月 \(7\)](#)
[2015年7月 \(22\)](#)
[2015年6月 \(5\)](#)

最新评论

1. Re:Learn Prolog Now
翻译 - 第一章 - 事实, 规则
和查询 - 第一节, 一些简单
的例子
学习!
--深蓝医生
2. Re:Learn Prolog Now
翻译 - 第一章 - 事实, 规则
和查询 - 第一节, 一些简单
的例子
翻译了这么多了, 而且每天一
篇, 不能望其项背啊。

Learn Prolog Now 翻译 - 第五章 - 数字运算 - 第二节, 数字运算与列表

内容提要

列表中的一些数字运算, 累加器

尾递归调用

列表中的一些数字运算, 累加器

关于数字运算最为重要的应用, 可能是获取一些数据结构体的一些有用事实, 比如列表。例如, 知道列表的长度是很有用的。我们将会给出一些使用列表和数字运算的例子。

一个列表的长度是多少? 这里有一个递归定义:

1. 空列表的长度为0.
2. 非空列表的长度为 $1 + \text{len}(T)$, 其中 $\text{len}(T)$ 是非空列表的尾部。

这个定义在Prolog中很容易实现, 以下是实现代码:

```
len([], 0).
```

```
len([_|T], N) :- len(T, X), N is X + 1.
```

这个谓词会如期望的运行, 比如:

```
?- len([a, b, c, d, e, [a, b], g], X).
```

```
X = 7
```

这是一个不错的程序: 很容易理解, 并且很书写很高效。但是还有其他一些求列表长度的方式。我们将会学习这种替代方式, 因为它会引入累加器的概念。如果你有使用其他编程语言的经验,

你可能已经知道使用变量保持中间结果的概念, 累加器就是Prolog中对应的思路。

如下是一个使用累加器计算列表长度的例子, 我们将定义一个谓词,

`accLen/3`, 有如下的参数:

```
accLen(List, Acc, Length)
```

这里的List就是我们要求解的列表, Length就是列表的长度 (是一个整数)。Acc是什么? 就是我们用于保存长度中间值的累加器 (所以也是一个整数)。我们的思路是, 如果我们调用这个

谓词, 将Acct初始化为0; 当递归对列表进行操作时, 每当找到一个头元素, 就将Acct加1, 直到列表为空; 当列表为空时, Acc就会保存列表的长度, 下面是代码:

```
accLen([_|T], A, L) :- Anew is A+1, accLen(T, Anew, L).
```

--Benjamin Yan

阅读排行榜

1. Learn Prolog Now 翻译 - 第三章 - 递归 - 第一节, 递归的定义(1168)
2. Learn Prolog Now 翻译 - 第一章 - 事实, 规则和查询 - 第一节, 一些简单的例子(1087)
3. Learn Prolog Now 翻译 - 第一章 - 事实, 规则和查询 - 第二节, Prolog语法介绍(781)
4. Haskell学习笔记二: 自定义类型(767)
5. Learn Prolog Now 翻译 - 第六章 - 列表补遗 - 第一节, 列表合并(753)

评论排行榜

1. Learn Prolog Now 翻译 - 第一章 - 事实, 规则和查询 - 第一节, 一些简单的例子(2)

推荐排行榜

1. Haskell学习笔记二: 自定义类型(1)
2. Learn Prolog Now 翻译 - 第三章 - 递归 - 第四节, 更多的实践和练习(1)

accLen([], A, A).

关于基础子句的定义, 将第二个参数和第三个参数进行了合一。为什么? 因为这个简单的合一是返回结果的良好方式。当达到了列表的底部, 累加器(第二个参数)持有了列表的长度值, 所以

所以将这个值通过合一赋予长度变量(第三个参数)。下面是一个例子的追踪, 可以清晰地看到当Prolog到达列表底部, 长度变量通过合一进行了赋值:

?- accLen([a, b, c], 0, L).

Call: (6) accLen([a, b, c], 0, _G499) ?

Call : (7) _G518 is 0 + 1 ?

Exit: (7) 1 is 0 + 1 ?

Call: (7) accLen([b, c], 1, _G499) ?

Call: (8) _G521 is 1+1 ?

Exit: (8) 2 is 1+1 ?

Call: (8) accLen([a], 2, _G499) ?

Call: (9) _G524 is 2+1 ?

Exit: (9) 3 is 2+1?

Call: (9) accLen([], 3, _G499) ?

Exit: (9) accLen([], 3, 3) ?

Exit: (8) accLen([c], 2, 3) ?

Exit: (7) accLen([b, c], 1, 3) ?

Exit: (6) accLen([a, b, c], 0, 3)?

最后, 我们可以定义一个谓词调用accLen, 并且给出累加器的初始值为0:

```
leng(List, Length) :- accLen(List, 0, Length).
```

所以, 我们可以进行如下的查询:

```
?- leng([a, b, c, d, e, [a, b], g], X).
```

```
X = 7
```

尾递归调用

累加器在Prolog中是很常用的(后面的章节会看到更多使用累加器的例子), 但是为什么会这样? accLen在哪个方面比len更好呢? 毕竟, accLen看上去更加复杂。答案就是因为accLen是

尾递归调用, 但len不是。在一个尾递归调用的程序里, 当递归到底底部的时候, 结果已经计算得出, 剩下所需要做的, 就是逐层返回。在一个不是尾递归调用的递归中, 一层的目标会等待

更里层的结果返回后, 再进行计算。为了更清楚地理解, 可以对比查询accLen([a, b, c], 0, L)的追踪, 和查询len([a, b, c], L) (如下所示):

?- len([a, b, c], L).

Call: (6) len([a, b, c], _G418) ?

Call: (7) len([b, c], _G481) ?

Call: (8) len([c], _G486) ?

Call: (9) len([], _G489) ?

Exit: (9) len([], 0) ?

Call: (9) _G486 is 0+1 ?

Exit: (9) 1 is 0+1 ?

Exit: (8) len([c], 1) ?

Call: (8) _G481 is 1+1 ?

Exit: (8) 2 is 1+1 ?

Exit: (7) len([b, c], 2) ?

Call: (7) _G418 is 2+1 ?

Exit: (7) 3 is 2+1 ?

Exit: (6) len([a, b, c], 3) ?

在accLen的查询追踪里, 当递归到底底部, accLen([], 3, _G449), 结果就已经计算完毕, 剩下只是回传上去。在len的查询追踪里面, 结果的计算依赖递归, 比如, len([b, c], _G481)的结果,

只能在完成len([c], _G489)的结果后才能进行计算。简而言之, 尾递归程序会有更少的中间变量回溯计算, 这使得递归会更高效。

分类: Prolog

标签: 列表, 数字运算, 尾递归, 累加器

好文要顶

关注我

收藏该文



seaman.kingfall

关注 - 1

粉丝 - 4

+加关注

0

0

« 上一篇: Learn Prolog Now 翻译 - 第五章 - 数字运算 - 第一节, Prolog中的数字运算

» 下一篇: Learn Prolog Now 翻译 - 第五章 - 数字运算 - 第三节, 整数的比较

posted on 2015-07-16 10:57 seaman.kingfall 阅读(500) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

努力加载评论框中...

【推荐】超50万C++/C#源码: 大型实时仿真组态图形源码

【活动】看雪2019安全开发者峰会, 共话安全领域焦点

【培训】Java程序员年薪40W, 他1年走了别人5年的路

最新新闻:

- 微信公开课聚焦“增长”: 墨迹天气小程序DAU环比增100%
 - 知否 | 太空垃圾如何清理? 卫星测试用鱼叉击中太空垃圾碎片
 - 一线 | “美团配送”品牌发布: 对外开放配送平台 共享配送能力
 - 苍蝇落在食物上会发生什么? 让我们说的仔细一点
 - 科学家研究板块构造变化对海洋含氧量影响
- » 更多新闻...

Copyright © seaman.kingfall
Powered by: .Text and ASP.NET
Theme by: .NET Monster