

deriv.scm

实现了如下功能：

1. 多参数的+, -, *, /以及 所有基本初等函数的导数

```
> (deriv '(log (* x (+ x 2))) 'x)
(/ (+ 2 (* 2 x)) (+ (* 2 x) (** x 2)))
```

2. 比较多的结果化简

```
> (deriv '(* x x x (+ x 2) 3 (- 4 x)) 'x)
(+ (* 72 (** x 2)) (+ (* 24 (** x 3)) (* -15 (** x 4))))
```

3. 如果结果为多项式，将其规约为标准形式 上面的两个例子已经展示了这个效果

4. 一些有用的工具函数 将值带入表达式：

```
> (eval-at '(+ x 2 (* y 5) y) '(x 2) '(y 5))
34
```

将运算结果保存为scheme中的函数：

```
> ((expr->lambda (deriv '(* x x x) 'x) '(x)) 4)
48
```

5. 简单的中缀表达式支持

```
> (S->infix (deriv '(* x (+ x 2) (- x 5)) 'x))
(-10 + ((-6 * x) + (3 * (x ** 2))))
```

由于时间原因，S->infix和infix->S都只实现了最简单的功能，没有去掉多余括号

6. future work: 原计划实现调用gnuplot对表达式画图、打印出表达式的LaTeX代码等功能，由于时间原因没有实现

具体的实现请参照代码，里面有不少注释，代码本身也是自解释的，就不在此赘述了。

杨至轩 2014 3 23