

## 16.38 LISTVECOPS: Vector operations on lists

Author: Eberhard Schrüfer

This package implements vector operations on lists.. Addition, multiplication, division, and exponentiation work elementwise. For example, after

```
A := {a1, a2, a3, a4};
B := {b1, b2, b3, b4};
```

$c*A$  will simplify to  $\{c*a1, \dots, c*a4\}$ ,  $A + B$  to  $\{a1+b1, \dots, a4+b4\}$ , and  $A*B$  to  $\{a1*b1, \dots, a4*b4\}$ . Linear operations work as expected:

```
c1*A + c2*B;

{a1*c1 + b1*c2,
 a2*c1 + b2*c2,
 a3*c1 + b3*c2,
 a4*c1 + b4*c2}
```

A division and an exponentiation example:

```
{a, b, c} / {3, g, 5};
```

$$\left\{ \frac{a}{3}, \frac{b}{g}, \frac{c}{5} \right\}$$

```
ws^3;
```

$$\left\{ \frac{a^3}{27}, \frac{b^3}{3}, \frac{c^3}{125} \right\}$$

The new operator  $*.$  (`ldot`) implements the dot product:

```
{a, b, c, d} *. {5, 7, 9, 11/d};
```

```
5*a + 7*b + 9*c + 11
```

For accessing list elements, the new operator `_ (lnth)` can be used instead of the `PART` operator:

```
l := {1, {2, 3}, 4}$
```

```
lnth(l, 3);
```

```
4
```

```
l _2*3;
```

```
{6, 9}
```

```
l _2 _2;
```

```
3
```

It can also be used to modify a list (unlike `PART`, which returns a modified list):

```
part(l, 2, 2) :=three;
```

```
{1, {2, three}, 4}
```

```
l;
```

```
{1, {2, 3}, 4}
```

```
l _ 2 _2 :=three;
```

```
three
```

```
l;
```

```
{1, {2, three}, 4}
```

Operators are distributed over lists:

```
a *. log b;
```

```
log(b1)*a1 + log(b2)*a2 + log(b3)*a3 + log(b4)*a4
```

```
df({sin x*y, x^3*cos y}, x, 2, y);
```

```
{ - sin(x), - 6*sin(y)*x}
```

```
int({sin x, cos x}, x);  
  
{ - cos(x), sin(x) }
```

By using the keyword **listproc**, an algebraic procedure can be declared to return a list:

```
listproc spat3(u, v, w);  
  begin scalar x, y;  
    x := u *. w;  
    y := u *. v;  
    return v*x - w*y  
  end;
```