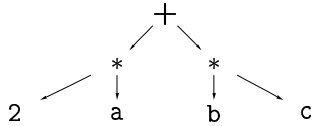


## Operator Notation II

### Operator Notation

Consider:  $2 * a + b * c$



To obtain that: use

```

:-op( 600, xfx, has )
      ↑      ↑      ↑
      precedence infix name
  
```

**Precedence:** describes the strength of binding of the operator. The higher the precedence of the given operator, the lower the strength of binding, and the higher up in the structure hierarchy the operator will appear. Of two operators competing for the same substructure, the one with the *lower* precedence will be assigned that structure as child.

**(In-)fix tag:** there are following tag types

```

xfx, xfy, yfx  infix
fx, fy         prefix  f denotes the operator,
xf, yf         postfix
  
```

*x* and *y* each denote what can be used as an argument for *f*. *x* accepts only structures as arguments whose principal functors have precedence smaller than that of *f*, *y* accepts structures as arguments whose principal functors have precedence not larger than *f*.

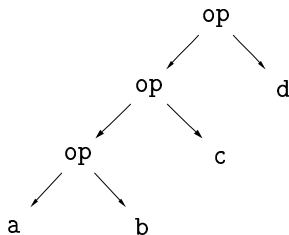
1

2

### Infix Example

**Infix tag meaning:** This makes e.g. an yfx operator *op* a left associative such that  $a \text{ op } b \text{ op } c \text{ op } d$  is interpreted as

$((a \text{ op } b) \text{ op } c) \text{ op } d$ , i.e.



3