

3e. Universal enveloping algebra, routines

See §3.1

The following routines are used for a standard ordering.

If another ordering is desired, then the lists XWlist and/or Zlist can be redefined.

```
In[ ]:= Clear[posXW, posZ]
posXW[xx_] := Block[{p}, p = Position[XWlist, xx]; If[p == {}, 0, p[[1]][[1]]]
posZ[xx_] := Block[{p}, p = Position[Zlist, xx];
  If[p == {}, 0, p[[1]][[1]]]
```

The multiplication in $U(\mathfrak{g})$ is implemented as NonCommutativeMultiply

```
In[ ]:= Unprotect[NonCommutativeMultiply ]
aa_** (bb_ + cc_) := aa** bb + aa** cc
(aa_ + bb_)** cc_ := aa** cc + bb** cc
aa_** nul := nul
nul** aa_ := nul
aa_** 0 := 0
0** aa_ := 0
(-aa_)** bb_ := -(aa** bb)
aa_** (-bb_) := -(aa** bb)
(ff_ aa_)** bb_ := ff (aa** bb) /; NumberQ[ff]
aa_** (ff_ bb_) := ff (aa** bb) /; NumberQ[ff]
xx_** yy_ := yy** xx + lb[xx, yy] /; 0 < posXW[yy] < posXW[xx]
xx_** yy_ := yy** xx + lb[xx, yy] /; 0 < posZ[yy] < posZ[xx]
Unprotect[Times]
ff_xx_** yy_ := ff yy** xx + ff lb[xx, yy] /; 0 < posZ[yy] < posZ[xx]
Protect[Times]
ff_** aa_ := ff aa /; NumberQ[ff]
aa_** ff_ := ff aa /; NumberQ[ff]
Protect[NonCommutativeMultiply ]
```

```
Out[ ]:= {NonCommutativeMultiply }
```

```
Out[ ]:= {Times}
```

```
Out[ ]:= {Times}
```

```
Out[ ]:= {NonCommutativeMultiply }
```

Checks

