Pensieve header: The Drinfel'd-Kohno algebra and associators.

Target: Implement DK[k, LW[...]]: b, Act.

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\2015-02"];
<< ../Projects/WKO4/FreeLie.m
```

```
FreeLie` implements / extends
 {*, +, **, $SeriesShowDegree, ⟨⟩, ∫, ≡, ad, Ad, adSeries, AllCyclicWords,
  AllLyndonWords, AllWords, ASeries, AW, b, BCH, BooleanSequence, BracketForm,
  BS, CC, Crop, CW, CWS, CWSeries, D, Deg, DegreeScale, DerivationSeries, div,
  EulerE, Exp, Inverse, j, J, JA, LieDerivation, LieMorphism, LieSeries, LS, LW,
  LyndonFactorization, New, RandomCWSeries, Randomizer, RandomLieSeries, RC, SeriesSolve,
  Support, tb, TopBracketForm, tr, UndeterminedCoefficients, Γ, ι, Λ, ħ, ⌐, ⌢}.
```

```
DK[_, 0] = 0;
DK /: DK[k_, x1_] + DK[k_, x2_] := DK[k, x1 + x2];
DK /: c_ * DK[k_, x_] := DK[k, Expand[c x]];
DK /: b[DK[k1_, c_ * x1_LW], DK[k2_, x2_]] :=
  Expand[c b[DK[k1, x1], DK[k2, x2]]];
DK /: b[DK[k1_, x1_LW], DK[k2_, c_ * x2_]] := Expand[c b[DK[k1, x1], DK[k2, x2]]];
DK /: b[DK[k1_, x1_Plus], DK[k2_, x2_]] := b[DK[k1, #], DK[k2, x2]] & /@ x1;
DK /: b[DK[k1_, x1_], DK[k2_, x2_Plus]] := b[DK[k1, x2], DK[k2, #]] & /@ x2;
DK /: b[DK[k_, x1_], DK[k_, x2_]] := DK[k, b[x1, x2]];
DK /: b[DK[k1_, x1_], DK[k2_, x2_]] /; k1 > k2 :=
  b[DK[k2, Expand[-x2], DK[k1, x1]]];
DK /: b[DK[k1_, LW@i1_], DK[k2_, LW@i2_]] /; k1 < k2 :=
  b[DK[k1, LW@i1], DK[k2, LW@i2]] = Which[
    i1 == i2, DK[k2, -b[LW@k1, LW@i2]],
    k1 == i2, DK[k2, -b[LW@i1, LW@i2]],
    True, 0
   ];
DK /: b[DK[k1_, w1_LW], DK[k2_, w2_LW]] /; Deg[w1] > 1 :=
  b[DK[k1, w1], DK[k2, w2]] =
   b @@ (b[DK[k1, #], DK[k2, w2]] & /@ LyndonFactorization[w1]);
DK /: b[DK[k1_, w1_LW], DK[k2_, w2_LW]] /; Deg[w2] > 1 :=
  b[DK[k1, w1], DK[k2, w2]] =
   b @@ (b[DK[k1, w1], DK[k2, #]] & /@ LyndonFactorization[w2]);
b[DK[k1_, ls1_LieSeries], DK[k2_, ls2_LieSeries]] /; k1 < k2 :=
  b[DK[k1, ls1], DK[k2, ls2]] = DK[k2, New[
     LieSeries[ser],
     ser[d_Integer] := ser[d] = (Sum[b[DK[k1, ls1[j]], DK[k2, ls2[d - j]]], {j, 1, d - 1}])[[2]]
    ]];
t[i_, j_] := DK[Max[i, j], LW@Min[i, j]];
Format[DK[k_, x_]] := x // LieMorphism[Table[LW@i → LW@t_{10 i+k}, {i, k - 1}]]
```

```
b[t[1, 2], t[1, 3]]
```

$$\mathrm{LS}\left[0, \overline{t_{13}\, t_{23}}, 0, \ldots\right]$$

```
b[t[1, 2], t[1, 3]] + b[t[1, 2], t[2, 3]]
```

$$0$$

```
σ[lft___, i_Integer, rgt___] := σ[lft, IntegerDigits[i], rgt];
x_Plus // s_σ := s[#] & /@ x;
DK[k_, LW@i_] // s_σ := Sum[t[α, β], {α, s[[i]]}, {β, s[[k]]}];
DK[k_, ls_LieSeries] // s_σ := DK[k, ls] // s =
```

```
t[1, 2] // σ[1, 34]
```

$$\text{LS}\left[\overline{t_{13}}, 0, 0, \ldots\right] + \text{LS}\left[\overline{t_{14}}, 0, 0, \ldots\right]$$

```
t[1, 2] // σ[4, 3]
```

$$\text{LS}\left[\overline{t_{34}}, 0, 0, \ldots\right]$$