Free Lie Algebras Routines

Lazy Evaluation Version

Pensieve header: A free-Lie calculator with lazy evaluation for series.

Global Definitions

```
$SeriesShowDegree = 3; $SeriesCompareDegree = 3;
```

Words and Lyndon Words

A Lyndon word is a word lexicographically smaller than all of its proper right factors; see http://katlas.math.toronto.edu/drorbn/AcademicPensieve/Projects/FreeLie/index.html

```
LyndonQ[AW[w_String]] := And @@ (
    OrderedQ[{w, #}] & /@ Table[StringDrop[w, i], {i, 1, StringLength[w] - 1}]
   );
AllWords[0, _List] = {AW[""]};
AllWords[n_ /; n > 0, ab_List] := AllWords[n, ab] = AW /@ Flatten[Outer[
      StringJoin[#1, #2] &,
      First /@ AllWords[n - 1, ab],
      ab
     ]];
AllLyndonWords[n_Integer, ab_List] := LW @@@ Select[AllWords[n, ab], LyndonQ];
AllLyndonWords[{n_}, ab_List] := Join @@ Table[AllLyndonWords[k, ab], {k, n}];
LyndonFactorization[LW[w_String] /; StringLength[w] == 1] := LW[w];
LyndonFactorization[LW[w_String] /; StringLength[w] > 1] := Module[
   {rf},
   rf = First[Sort[Table[StringDrop[w, i], {i, 1, StringLength[w] - 1}]]];
   LW /@ {StringDrop[w, -StringLength[rf]], rf}
  ];
LW[s_Symbol] := LW[ToString[s]];
LW[LW[w_]] := LW[w];
LW /: LW[x_] ≤ LW[y_] := OrderedQ[{x, y}];
LW /: x_LW ≥ y_LW := y ≤ x;
LW /: x_LW > y_LW := ! (x ≤ y);
LW /: x_LW < y_LW := ! (y ≤ x);
Format[LW[w_], StandardForm] := Defer[〈w〉];
〈w__〉 := LW[w];
LW[is__Integer] := LW[
   StringJoin @@ (StringTake["1234567890abcdefghijklmnopqrstuvwxyz", {#}] & /@ {is})];
Deg[LW[x_]] := StringLength[x];
```

```
{LyndonQ[AW@"abba"], LyndonQ[AW@"ababb"]}
```

```
{False, True}
```

```
{AllWords[3, {"1", "2"}], AllLyndonWords[{3}, {"1", "2"}]}
```

```
{{AW[111], AW[112], AW[121], AW[122], AW[211], AW[212], AW[221], AW[222]},
 {⟨1⟩, ⟨2⟩, ⟨12⟩, ⟨112⟩, ⟨122⟩}}
```

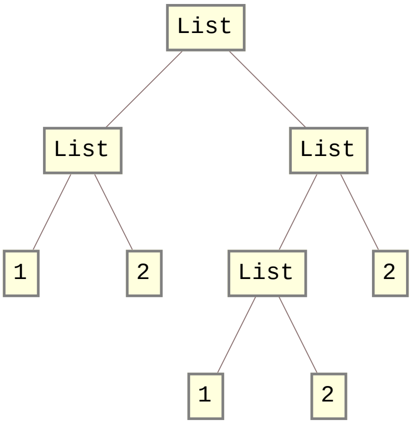
```
Table[Length[AllLyndonWords[k, {"1", "2"}]], {k, 10}]
```

```
{2, 1, 2, 3, 6, 9, 18, 30, 56, 99}
```

```
Table[Length[AllLyndonWords[k, {"1", "2", "3"}]], {k, 10}]
```

```
{3, 3, 8, 18, 48, 116, 312, 810, 2184, 5880}
```

```
TreeForm[LW["12122"] //. w_LW ⧴ LyndonFactorization[w] /. LW[w_] ⧴ w]
```



The Bracket for Lie Elements

```
b[0, _] = 0; b[_, 0] = 0;
b[c_ * (x_AW | x_LW), y_] := Expand[c b[x, y]];
b[x_, c_ * (y_AW | y_LW)] := Expand[c b[x, y]];
b[x_Plus, y_] := b[#, y] & /@ x;
b[x_, y_Plus] := b[x, #] & /@ y;
b[w_LW, z_LW] := LWAdjoint[w][z];
ad[x_][y_] := b[x, y];
```

```
LWAdjoint[w_] := LWAdjoint[w] = Module[{u},
    u = Unique[LWAct];
    u[z_] := u[z] = Which[
       w === z, 0,
       z < w, Expand[-b[z, w]],
       Deg[w] == 1, LW[First[w] <> First[z]],
       True, Module[{x, y},
        {x, y} = LyndonFactorization[w];
        If[y ≥ z,
         LW[First[w] <> First[z]],
         b[x, LWAdjoint[y][z]] + b[LWAdjoint[x][z], y]
        ]
       ]
      ];
    u
   ];
```

```
b[LW["112"], LW["122"]]
```

```
⟨112122⟩ + ⟨112212⟩
```

```
Outer[b, AllLyndonWords[{3}, {"1", "2"}],
  AllLyndonWords[{3}, {"1", "2"}]] // MatrixForm
```

```
⎛    0       ⟨12⟩     ⟨112⟩          ⟨1112⟩                ⟨1122⟩          ⎞
⎜  -⟨12⟩      0      -⟨122⟩         -⟨1122⟩               -⟨1222⟩          ⎟
⎜ -⟨112⟩    ⟨122⟩       0           -⟨11212⟩               ⟨12122⟩         ⎟
⎜-⟨1112⟩   ⟨1122⟩   ⟨11212⟩             0           ⟨112122⟩ + ⟨112212⟩   ⎟
⎝-⟨1122⟩   ⟨1222⟩  -⟨12122⟩  -⟨112122⟩ - ⟨112212⟩             0            ⎠
```

```
Union[Flatten[Outer[(b[#1, #2] + b[#2, #1]) &,
   AllLyndonWords[{6}, {"1", "2"}], AllLyndonWords[{6}, {"1", "2"}]
  ]]]
```

```
{0}
```

```
Outer[(b[#1, b[#2, #3]] + b[#2, b[#3, #1]] + b[#3, b[#1, #2]]) &,
   AllLyndonWords[{5}, {"1", "2"}],
   AllLyndonWords[{5}, {"1", "2"}], AllLyndonWords[{5}, {"1", "2"}]
  ] // Flatten //
 Union
```

```
{0}
```

LieSeries

```
LieSeries[ser_Symbol][{dd_Integer}] := LS @@ Table[ser[d], {d, dd}];
LieSeries[ser_Symbol][e___] := ser[e];
Format[s_LieSeries, StandardForm] := s[{$SeriesShowDegree}];
ShowLieSeries[d_Integer][s_LieSeries] := s[{d}];
MakeLieSeries[s_LieSeries] := s;
MakeLieSeries[expr_] :=
  MakeLieSeries[expr] = MakeLieSeries[Unique[MakeLieSeries], expr];
MakeLieSeries[ser_Symbol, expr_] := (
   ser[] = Hold[MakeLieSeries[ser, expr]];
   ser[d_Integer] := ser[d] = Expand[expr /. w_LW /; Deg[w] ≠ d → 0];
   LieSeries[ser]
  );
s1_LieSeries ≡ s2_LieSeries :=
  And @@ ((s1[#] == s2[#]) & /@ Range[$SeriesCompareDegree]);
```

```
Print /@ {ts1 = ⟨"1122"⟩ // MakeLieSeries, ts1[], ts1 /@ Range[6]};
```

```
LS[0, 0, 0]

Hold[MakeLieSeries[MakeLieSeries$554, ⟨1122⟩]]

{0, 0, 0, ⟨1122⟩, 0, 0}
```

```
AddLieSeries[ss___LieSeries] := AddLieSeries[ss] = Module[{ser},
    ser = Unique[AddLieSeries];
    ser[] = Hold[AddLieSeries[ss]];
    ser[d_Integer] := ser[d] = Plus @@ ((#[d]) & /@ {ss});
    LieSeries[ser]
   ];
ScaleLieSeries[c_, s_LieSeries] := ScaleLieSeries[c, s] = Module[{ser},
    ser = Unique[ScaleLieSeries];
    ser[] = Hold[ScaleLieSeries[ss]];
    ser[d_Integer] := ser[d] = Expand[c * s[d]];
    LieSeries[ser]
   ];
b[s_LieSeries, y_] := b[s, MakeLieSeries[y]];
b[x_, s_LieSeries] := b[MakeLieSeries[x], s];
```

```
b[s1_LieSeries, s2_LieSeries] := b[s1, s2] = Module[{ser},
    ser = Unique[b];
    ser[] = Hold[b[s1, s2]];
    ser[d_Integer] := ser[d] = Sum[
       b[s1[k], s2[d - k]],
       {k, 1, d - 1}
      ];
    LieSeries[ser]
   ];
b[s_LieSeries, y_] := b[s, MakeLieSeries[y]];
b[x_, s_LieSeries] := b[MakeLieSeries[x], s];
```

```
{ts2 = ⟨"122"⟩ + ⟨"11122"⟩ // MakeLieSeries, ts3 = b[ts1, ts2], ts3[], ts3 /@ Range[10]}
```

```
{LieSeries[MakeLieSeries$545], LieSeries[b$547],
 Hold[b[LieSeries[MakeLieSeries$543], LieSeries[MakeLieSeries$545]]],
 {0, 0, 0, 0, 0, 0, ⟨1122122⟩, 0, -⟨111221122⟩, 0}}
```

```
LieSeries /: EulerE[s_LieSeries] := Module[{ser},
   ser = Unique[EulerE];
   ser[] = Hold[EulerE[s]];
   ser[d_Integer] := ser[d] = Expand[d * s[d]];
   LieSeries[ser]
  ];
```

```
{ts4 = EulerE[ts3], ts4[], ts4 /@ Range[10]}
```

```
{LieSeries[EulerE$554], Hold[EulerE[LieSeries[b$547]]],
 {0, 0, 0, 0, 0, 0, 7 ⟨1122122⟩, 0, -9 ⟨111221122⟩, 0}}
```

adPower, adSeries, and Ad

```
adPower[0, x_LieSeries][ψ_LieSeries] := adPower[0, x][ψ] = Module[{ser},
    ser = Unique[adPower];
    ser[] = Hold[adPower[0, x][ψ]];
    ser[d_Integer] := ser[d] = ψ[d];
    LieSeries[ser]
   ];
adPower[n_Integer, x_LieSeries][ψ_LieSeries] := adPower[n, x][ψ] = Module[{ser},
    ser = Unique[adPower];
    ser[] = Hold[adPower[n, x][ψ]];
    ser[d_Integer] := ser[d] = b[x, adPower[n - 1, x][ψ]][d];
    LieSeries[ser]
   ];
adSeries[f_, x_LieSeries][ψ_LieSeries] := adSeries[f, x][ψ] = Module[{ser},
    ser = Unique[adSeries];
    ser[] = Hold[adSeries[f, x][ψ]];
    ser[d_Integer] := ser[d] = Module[{c},
       Expand[Sum[
         c = SeriesCoefficient[f, {ad, 0, k}];
         If[c == 0, 0, c * adPower[k, x][ψ][d]],
         {k, 0, d - 1}
        ]]
      ];
    LieSeries[ser]
   ];
adSeries[f_, x_][ψ_] := adSeries[f, MakeLieSeries[x]][MakeLieSeries[ψ]];
Ad[x_] := adSeries[E^(-ad), x];
```

```
{xs = MakeLieSeries[LW["x"]], ys = MakeLieSeries[LW["y"]],
 ts5 = adPower[0, xs][ys], ts5[], ts5 /@ Range[5]}
```

```
{LieSeries[MakeLieSeries$98], LieSeries[MakeLieSeries$99], LieSeries[adPower$101],
 Hold[adPower[0, LieSeries[MakeLieSeries$98]][LieSeries[MakeLieSeries$99]]],
 {⟨y⟩, 0, 0, 0, 0}}
```

```
adPower[3, xs][ys] /@ Range[5]
```

```
{0, 0, 0, ⟨xxxy⟩, 0}
```

```
{adSeries[E^(-ad), xs][ys] /@ Range[5], adSeries[E^(-ad), ys][xs] /@ Range[5]}
```

$$\left\{\left\{\langle y\rangle, -\langle xy\rangle, \frac{\langle xxy\rangle}{2}, -\frac{\langle xxxy\rangle}{6}, \frac{\langle xxxxy\rangle}{24}\right\}, \left\{\langle x\rangle, \langle xy\rangle, \frac{\langle xyy\rangle}{2}, \frac{\langle xyyy\rangle}{6}, \frac{\langle xyyyy\rangle}{24}\right\}\right\}$$

```
Ad[xs][ys][5]
```

$$\frac{\langle xxxxy\rangle}{24}$$

```
Ad[xs][ys][]
```

```
Hold[adSeries[e^-ad, LieSeries[MakeLieSeries$98]][LieSeries[MakeLieSeries$99]]]
```

LieDerivation, DerivationPower, DerivationSeries

```
LieDerivation[der_][es___] := der[es];
LieDerivation[rules_List] :=
  LieDerivation[rules] = LieDerivation[Unique[LieDerivation], rules];
LieDerivation[der_Symbol, rules_List] := (
   der[] = Hold[LieDerivation[der, rules]];
   (der[w_LW] /; Deg[w] ⩵ 1) :=
    (der[w] = MakeLieSeries[w /. Append[rules, _LW → 0]]);
   der[w_LW] := der[w] = Module[{x, y},
      {x, y} = LyndonFactorization[w];
      AddLieSeries[b[der[x], y], b[x, der[y]]]
     ];
   der[s_LieSeries] := der[s] = Module[{ser},
      ser = Unique[LieDerivationOnLieSeries];
      ser[] = Hold[der[s]];
      ser[d_] := ser[d] = Sum[
         der[s[k]][d],
         {k, 1, d}
        ];
      LieSeries[ser]
     ];
   der[as_ASeries] := der[as] = Module[{ser},
      ser = Unique[LieDerivationOnASeries];
      ser[] = Hold[der[as]];
      ser[d_] := ser[d] = Sum[
         Expand[as[k] /. AW[w_] ⧴ Sum[
             NonCommutativeMultiply[
              AW[StringTake[w, j - 1]],
              ι[der[LW[StringTake[w, {j}]]][d - k + 1]],
              AW[StringDrop[w, j]]
             ],
             {j, k}
            ]],
         {k, 1, d}
        ];
      ASeries[ser]
     ];
   der[cws_CWSeries] := der[cws] = Module[{ser},
      ser = Unique[LieDerivationOnCWSeries];
      ser[] = Hold[der[cws]];
      ser[d_] := ser[d] = Sum[
         Expand[cws[k] /. CW[w_] ⧴ Sum[
             tr[NonCommutativeMultiply[
               AW[StringTake[w, j - 1]],
               ι[der[LW[StringTake[w, {j}]]][d - k + 1]],
               AW[StringDrop[w, j]]
              ]],
             {j, k}
            ]],
         {k, 1, d}
        ];
      CWSeries[ser]
     ];
   der[expr_][d_] := Expand[expr /. {w_LW ⧴ der[w][d], s_LieSeries ⧴ der[s][d]}];
   LieDerivation[der]
  );
```

```
Print /@ {
   ld1 = LieDerivation[{〈1〉 → b[〈3〉, 〈1〉]}],
   ld1[],
   (# → ld1[#][{4}]) & /@ AllLyndonWords[{3}, {"1", "2"}],
   (〈"112"〉 // ld1 // ld1)[{5}]
  };
```

```
LieDerivation[LieDerivation$103]
```

```
Hold[LieDerivation[LieDerivation$103, {〈1〉 → -〈13〉}]]
```

```
{〈1〉 → LS[0, -〈13〉, 0, 0], 〈2〉 → LS[0, 0, 0, 0], 〈12〉 → LS[0, 0, -〈132〉, 0],
 〈112〉 → LS[0, 0, 0, -〈1132〉 + 〈1213〉], 〈122〉 → LS[0, 0, 0, -〈1322〉]}
```

```
LS[0, 0, 0, 0, 〈11332〉 - 〈12133〉 + 2 〈13132〉]
```

```
DerivationPower[0, der_LieDerivation][ψ_LieSeries] :=
  DerivationPower[0, der][ψ] = Module[{ser},
    ser = Unique[DerivationPower];
    ser[] = Hold[DerivationPower[0, der][ψ]];
    ser[d_Integer] := ser[d] = ψ[d];
    LieSeries[ser]
   ];
DerivationPower[n_Integer, der_LieDerivation][ψ_LieSeries] :=
  DerivationPower[n, x][ψ] = Module[{ser},
    ser = Unique[DerivationPower];
    ser[] = Hold[DerivationPower[n, der][ψ]];
    ser[d_Integer] := ser[d] = der[DerivationPower[n - 1, der][ψ]][d];
    LieSeries[ser]
   ];
DerivationSeries[__][0] = 0;
DerivationSeries[f_, ld_LieDerivation][ψ_LieSeries] :=
  DerivationSeries[f, ld][ψ] = Module[{ser},
    ser = Unique[DerivationSeries];
    ser[] = Hold[DerivationSeries[f, ld][ψ]];
    ser[d_Integer] := ser[d] = Module[{c},
       Expand[Sum[
         c = SeriesCoefficient[f, {der, 0, k}];
         If[c == 0, 0, c * DerivationPower[k, ld][ψ][d]],
         {k, 0, d}
        ]]
      ];
    LieSeries[ser]
   ];
DerivationExp[ld_LieDerivation] := DerivationSeries[E^der, ld];
```

```
〈"112"〉 // MakeLieSeries // DerivationExp[LieDerivation[{〈1〉 → b[〈3〉, 〈1〉]}]] //
 ShowLieSeries[6]
```

$$\mathrm{LS}\left[0,\ 0,\ \langle 112\rangle,\ -\langle 1132\rangle+\langle 1213\rangle,\ \frac{\langle 11332\rangle}{2}-\frac{\langle 12133\rangle}{2}+\langle 13132\rangle,\ -\frac{\langle 113332\rangle}{6}+\frac{\langle 121333\rangle}{6}-\frac{\langle 131332\rangle}{2}+\frac{\langle 132133\rangle}{2}\right]$$

```
〈"122"〉 // MakeLieSeries // DerivationExp[LieDerivation[{〈1〉 → b[〈3〉, 〈1〉]}]] //
 ShowLieSeries[6]
```

$$\mathrm{LS}\left[0,\ 0,\ \langle 122\rangle,\ -\langle 1322\rangle,\ \frac{\langle 13322\rangle}{2},\ -\frac{\langle 133322\rangle}{6}\right]$$

LieMorphism

```
LieMorphism[mor_][es___] := mor[es];
LieMorphism[rules_List] :=
  LieMorphism[rules] = LieMorphism[Unique[LieMorphism], rules];
LieMorphism[mor_Symbol, rules_List] := (
   mor[] = Hold[LieMorphism[mor, rules]];
   (mor[w_LW] /; Deg[w] == 1) := (mor[w] = MakeLieSeries[w /. rules]);
   mor[w_LW] := (mor[w] = b @@ (mor /@ LyndonFactorization[w]));
   mor[AW[""]] = MakeASeries[AW[""]];
   (mor[AW[w_]] /; StringLength[w] == 1) :=
    (mor[w] = ι[MakeLieSeries[LW[w] /. rules]]);
   mor[AW[w_]] := mor[w] = Module[{w1, w2},
      w1 = StringTake[w, Floor[StringLength[w] / 2]];
      w2 = StringDrop[w, Floor[StringLength[w] / 2]];
      (mor[AW[w1]]) ** (mor[AW[w2]])
     ];
   mor[CW[w_]] := tr[mor[AW[w]]];
   mor[s_LieSeries] := mor[s] = Module[{ser},
      ser = Unique[LieMorphismOnLieSeries];
      ser[] = Hold[mor[s]];
      ser[d_] := ser[d] = Sum[
         mor[s[k]][d],
         {k, 1, d}
        ];
      LieSeries[ser]
     ];
   mor[cws_CWSeries] := mor[cws] = Module[{ser},
      ser = Unique[LieMorphismOnCWSeries];
      ser[] = Hold[mor[s]];
      ser[d_] := ser[d] = Sum[
         mor[cws[k]][d],
         {k, 1, d}
        ];
      CWSeries[ser]
     ];
   mor[expr_][d_] := Expand[expr /. (w_LW | w_AW | w_CW) :> mor[w][d]];
   LieMorphism[mor]
  );
```

```
Print /@ {
   lm0 = LieMorphism[{LW["x"] → LW["y"]}],
   LW["x"] // lm0,
   AW["x"] // lm0,
   CW["x"] // lm0};
```

```
LieMorphism[LieMorphism$101]

LS[⟨y⟩, 0, 0]

AS[0, AW[y], 0, 0]

CWS[CW[y], 0, 0]
```

```
Print /@ {
   lm1 = LieMorphism[{LW["x"] → Ad[LW["y"]][LW["x"]]}],
   lm1[],
   lm1[LW["y"]],
   lm1[LW["x"]],
   lm1[LW["x"]][4],
   lm1[⟨"xxy"⟩],
   lm1[⟨"xxy"⟩][8],
   lm1[AW["x"]],
   lm1[CW["x"]]
  };
```

```
LieMorphism[LieMorphism$557]

                                                         ⟨xyy⟩
Hold[LieMorphism[LieMorphism$557, {⟨x⟩ → LS[⟨x⟩, ⟨xy⟩, ─────]}]]
                                                           2

LS[⟨y⟩, 0, 0]

                 ⟨xyy⟩
LS[⟨x⟩, ⟨xy⟩, ─────]
                   2

⟨xyyy⟩
──────
  6

LS[0, 0, ⟨xxy⟩]

⟨xxyyyyyy⟩   ⟨xyxyyyyy⟩   ⟨xyyxyyyy⟩
────────── + ────────── + ──────────
   120           30           24

                                    AW[xyy]              AW[yyx]
AS[0, AW[x], AW[xy] - AW[yx], ─────── - AW[yxy] + ───────]
                                       2                    2

CWS[CW[x], 0, 0]
```

StableApply

```
StableApply[mor_LieMorphism, (type : (LieSeries | ASeries | CWSeries))[s_]] := (
   StableApply[mor, type[s]] = Module[{ser},
     ser = Unique[StableApply];
     ser[] = Hold[StableApply[mor, type[s]]];
     ser[d_] := ser[d] = Nest[mor, type[s], d][d];
     type[ser]
    ]
  );
```

BCH

```
BCHBase = Module[{bch},
   bch = Unique["BCHBase"];
   bch[] = Hold[BCHBase];
   bch[1] = ⟨"x"⟩ + ⟨"y"⟩;
   bch[d_Integer] := bch[d] = Expand[Plus[
        adSeries[E^(-ad), MakeLieSeries[⟨"y"⟩]][MakeLieSeries[⟨"x"⟩]][d],
        -adSeries[(1 - E^(-ad)) / ad - 1, LieSeries[bch]][EulerE[LieSeries[bch]]][d]
      ] / d];
   LieSeries[bch]
  ];
BCH[x_, y_] := LieMorphism[{LW["x"] → x, LW["y"] → y}][BCHBase];
```

```
{BCHBase, BCHBase[], BCHBase[8]}
```

$$\left\{\text{BCHBase3}\left[\langle\text{x}\rangle+\langle\text{y}\rangle,\ \frac{\langle\text{xy}\rangle}{2},\ \frac{\langle\text{xxy}\rangle}{12}+\frac{\langle\text{xyy}\rangle}{12}\right],\ \text{Hold}[\text{BCHBase}],\ \frac{\langle\text{xxxxxxyy}\rangle}{60\,480}-\frac{\langle\text{xxxxxyxy}\rangle}{15\,120}-\frac{\langle\text{xxxxxyyy}\rangle}{10\,080}+\frac{\langle\text{xxxxyxxy}\rangle}{20\,160}-\frac{\langle\text{xxxxyxyy}\rangle}{20\,160}+\frac{\langle\text{xxxxyyxy}\rangle}{2520}+\frac{23\,\langle\text{xxxxyyyy}\rangle}{120\,960}+\frac{\langle\text{xxxyxxyy}\rangle}{4032}-\frac{\langle\text{xxxyxyxy}\rangle}{10\,080}+\frac{13\,\langle\text{xxxyxyyy}\rangle}{30\,240}+\frac{\langle\text{xxxyyxyy}\rangle}{20\,160}-\frac{\langle\text{xxxyyyxy}\rangle}{3024}-\frac{\langle\text{xxxyyyyy}\rangle}{10\,080}+\frac{\langle\text{xxyxyxyy}\rangle}{2520}-\frac{\langle\text{xxyxyyyy}\rangle}{4032}-\frac{\langle\text{xxyyxyyy}\rangle}{10\,080}+\frac{\langle\text{xxyyyyyy}\rangle}{60\,480}\right\}$$

$$\left\{\text{LieSeries}[\text{BCHBase3}],\ \text{Hold}[\text{BCHBase}],\ \frac{\langle\text{xxxxxxyy}\rangle}{60\,480}-\frac{\langle\text{xxxxxyxy}\rangle}{15\,120}-\frac{\langle\text{xxxxxyyy}\rangle}{10\,080}+\frac{\langle\text{xxxxyxxy}\rangle}{20\,160}-\frac{\langle\text{xxxxyxyy}\rangle}{20\,160}+\frac{\langle\text{xxxxyyxy}\rangle}{2520}+\frac{23\,\langle\text{xxxxyyyy}\rangle}{120\,960}+\frac{\langle\text{xxxyxxyy}\rangle}{4032}-\frac{\langle\text{xxxyxyxy}\rangle}{10\,080}+\frac{13\,\langle\text{xxxyxyyy}\rangle}{30\,240}+\frac{\langle\text{xxxyyxyy}\rangle}{20\,160}-\frac{\langle\text{xxxyyyxy}\rangle}{3024}-\frac{\langle\text{xxxyyyyy}\rangle}{10\,080}+\frac{\langle\text{xxyxyxyy}\rangle}{2520}-\frac{\langle\text{xxyxyyyy}\rangle}{4032}-\frac{\langle\text{xxyyxyyy}\rangle}{10\,080}+\frac{\langle\text{xxyyyyyy}\rangle}{60\,480}\right\}$$

```
{BCH[LW["y"], LW["z"]], BCH[LW["y"], LW["z"]][6]}
```

$$\left\{\text{LieSeries}[\text{LieMorphismOnLieSeries\$101}],\ -\frac{\langle\text{yyyyzz}\rangle}{1440}+\frac{\langle\text{yyyzyz}\rangle}{720}+\frac{\langle\text{yyyzzz}\rangle}{360}+\frac{\langle\text{yyzyzz}\rangle}{240}-\frac{\langle\text{yyzzzz}\rangle}{1440}\right\}$$

```
{
  t1 = BCH[LW["x"], BCH[LW["y"], LW["z"]]],
  t2 = BCH[BCH[LW["x"], LW["y"]], LW["z"]],
  t1 == t2,
  Table[t1[d] == t2[d], {d, 10 }]
 } // Timing
```

$$\Big\{7.987,\ \Big\{\mathrm{LS}\Big[\langle x\rangle+\langle y\rangle+\langle z\rangle,\ \frac{\langle xy\rangle}{2}+\frac{\langle xz\rangle}{2}+\frac{\langle yz\rangle}{2},$$
$$\frac{\langle xxy\rangle}{12}+\frac{\langle xxz\rangle}{12}+\frac{\langle xyy\rangle}{12}+\frac{\langle xyz\rangle}{3}+\frac{\langle xzy\rangle}{6}+\frac{\langle xzz\rangle}{12}+\frac{\langle yyz\rangle}{12}+\frac{\langle yzz\rangle}{12}\Big],\ \mathrm{LS}\Big[\langle x\rangle+\langle y\rangle+\langle z\rangle,$$
$$\frac{\langle xy\rangle}{2}+\frac{\langle xz\rangle}{2}+\frac{\langle yz\rangle}{2},\ \frac{\langle xxy\rangle}{12}+\frac{\langle xxz\rangle}{12}+\frac{\langle xyy\rangle}{12}+\frac{\langle xyz\rangle}{3}+\frac{\langle xzy\rangle}{6}+\frac{\langle xzz\rangle}{12}+\frac{\langle yyz\rangle}{12}+\frac{\langle yzz\rangle}{12}\Big],$$
$$\mathrm{LS}\Big[\langle x\rangle+\langle y\rangle+\langle z\rangle,\ \frac{\langle xy\rangle}{2}+\frac{\langle xz\rangle}{2}+\frac{\langle yz\rangle}{2},$$
$$\frac{\langle xxy\rangle}{12}+\frac{\langle xxz\rangle}{12}+\frac{\langle xyy\rangle}{12}+\frac{\langle xyz\rangle}{3}+\frac{\langle xzy\rangle}{6}+\frac{\langle xzz\rangle}{12}+\frac{\langle yyz\rangle}{12}+\frac{\langle yzz\rangle}{12}\Big] == \mathrm{LS}\Big[\langle x\rangle+\langle y\rangle+\langle z\rangle,$$
$$\frac{\langle xy\rangle}{2}+\frac{\langle xz\rangle}{2}+\frac{\langle yz\rangle}{2},\ \frac{\langle xxy\rangle}{12}+\frac{\langle xxz\rangle}{12}+\frac{\langle xyy\rangle}{12}+\frac{\langle xyz\rangle}{3}+\frac{\langle xzy\rangle}{6}+\frac{\langle xzz\rangle}{12}+\frac{\langle yyz\rangle}{12}+\frac{\langle yzz\rangle}{12}\Big],$$
$$\{\mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True},\ \mathrm{True}\}\Big\}\Big\}$$

AW, ASeries, ι, σ

```
Unprotect[NonCommutativeMultiply];
x_ ** 0 = 0; 0 ** y_ = 0;
(c_ * x_AW) ** y_ := Expand[c (x ** y)];
x_ ** (c_ * y_AW) := Expand[c (x ** y)];
x_Plus ** y_ := (# ** y) & /@ x;
x_ ** y_Plus := (x ** #) & /@ y;
Deg[AW[w_]] := StringLength[w];
AW[AW[w_]] := AW[w];
AW[w1_String] ** AW[w2_String] := AW[w1 <> w2];
b[w_AW, z_AW] := w ** z - z ** w;
```

```
ASeries[ser_Symbol][{dd_Integer}] := AS @@ Table[ser[d], {d, 0, dd}];
ASeries[as_Symbol][es___] := as[es];
Format[s_ASeries, StandardForm] := s[{$SeriesShowDegree}];
MakeASeries[as_CWSeries] := as;
MakeASeries[expr_] :=
  MakeASeries[expr] = MakeCWSeries[Unique[MakeASeries], expr];
MakeASeries[ser_Symbol, expr_] := (
   ser[] = Hold[MakeASeries[ser, expr]];
   ser[d_Integer] := ser[d] = Expand[expr /. w_AW /; Deg[w] ≠ d → 0];
   ASeries[ser]
  );
(s1_ASeries ** s2_ASeries) := (s1 ** s2) = Module[{ser},
    ser = Unique[NonCommutativeMultiply];
    ser[] = Hold[s1 ** s2];
    ser[d_Integer] := ser[d] = Sum[
       s1[k] ** s2[d - k],
       {k, 0, d}
      ];
    ASeries[ser]
   ];
```

```
ι[w_LW] /; Deg[w] == 1 := AW @@ w;
ι[w_LW] := ι[w] = b @@ (ι /@ LyndonFactorization[w]);
ι[expr_] := Expand[expr /. w_LW :→ ι[w]];
ι[ls_LieSeries] := ι[ls] = Module[{as},
    as = Unique[ι];
    as[] = Hold[ι[ls]];
    as[d_] := as[d] = ι[ls[d]];
    ASeries[as]
   ];
```

```
ι[BCHBase[3]]
```

```
AW[xxy]   AW[xyx]   AW[xyy]   AW[yxx]   AW[yxy]   AW[yyx]
------- - ------- + ------- + ------- - ------- + -------
  12         6        12        12         6        12
```

```
{as = ι[BCHBase], as[5]}
```

```
                    AW[xxxxy]   AW[xxxyx]   AW[xxxyy]   AW[xxyxx]   AW[xxyxy]
{ASeries[ι$605], - --------- + --------- + --------- - --------- - --------- -
                       720         180         180         120         120
  AW[xxyyx]   AW[xxyyy]   AW[xyxxx]   AW[xyxxy]   AW[xyxyx]   AW[xyxyy]
  --------- + --------- + --------- - --------- + --------- - --------- -
     120         180         180         120          30         120
  AW[xyyxx]   AW[xyyxy]   AW[xyyyx]   AW[xyyyy]   AW[yxxxx]   AW[yxxxy]
  --------- - --------- + --------- - --------- - --------- + --------- -
     120         120         180         720         720         180
  AW[yxxyx]   AW[yxxyy]   AW[yxyxx]   AW[yxyxy]   AW[yxyyx]   AW[yxyyy]
  --------- - --------- - --------- + --------- - --------- + --------- +
     120         120         120          30         120         180
  AW[yyxxx]   AW[yyxxy]   AW[yyxyx]   AW[yyxyy]   AW[yyyxx]   AW[yyyxy]   AW[yyyyx]
  --------- - --------- - --------- - --------- + --------- + --------- - ---------}
     180         120         120         120         180         180         720
```

```
σ[y_LW, w_LW] /; Deg[y] == 1 := σ[y, w] = Which[
    y === w, AW[""],
    Deg[w] === 1, 0,
    True, Module[{w1, w2},
     {w1, w2} = LyndonFactorization[w];
     ι[w1] ** σ[y, w2] - ι[w2] ** σ[y, w1]
    ]
   ];
σ[y_, ls_LieSeries] := σ[y, ls] = Module[{as},
    as = Unique[σ];
    as[] = Hold[σ[y, ls]];
    as[d_] := as[d] = σ[LW[y], ls[d + 1]];
    ASeries[as]
   ];
σ[y_, expr_] := Expand[expr /. w_LW ⧴ σ[LW[y], w]];
```

```
(# -> σ[1, #]) & /@ AllLyndonWords[{5}, {"1", "2"}]
```

```
{⟨1⟩ → AW[], ⟨2⟩ → 0, ⟨12⟩ → -AW[2], ⟨112⟩ → -2 AW[12] + AW[21], ⟨122⟩ → AW[22],
 ⟨1112⟩ → -3 AW[112] + 3 AW[121] - AW[211], ⟨1122⟩ → 2 AW[212] - AW[221],
 ⟨1222⟩ → -AW[222], ⟨11112⟩ → -4 AW[1112] + 6 AW[1121] - 4 AW[1211] + AW[2111],
 ⟨11122⟩ → -AW[1122] + 4 AW[1212] - AW[1221] - 2 AW[2121] + AW[2211],
 ⟨11212⟩ → -AW[1122] + 4 AW[1212] - AW[1221] - 3 AW[2112] + AW[2121],
 ⟨11222⟩ → -2 AW[1222] + 3 AW[2122] - 3 AW[2212] + AW[2221],
 ⟨12122⟩ → 2 AW[1222] - 3 AW[2122] + AW[2212], ⟨12222⟩ → AW[2222]}
```

```
{σ["x", BCHBase][5], σ["y", BCHBase][5]}
```

```
   AW[yxxxy]   AW[yxxyx]   AW[yxxyy]   AW[yxyxx]   AW[yxyxy]   AW[yxyyx]   AW[yxyyy]
{- ───────── + ───────── + ───────── - ───────── - ───────── + ───────── - ───────── +
      360         240         240         360          60         240         360
   AW[yyxxx]   AW[yyxxy]   AW[yyxyx]   AW[yyxyy]   AW[yyyxx]   AW[yyyxy]   AW[yyyyx]
   ───────── + ───────── + ───────── + ───────── - ───────── - ───────── + ─────────,
     1440         240         240         240         360         360        1440
   AW[xxxxy]   AW[xxxyx]   AW[xxxyy]   AW[xxyxx]   AW[xxyxy]   AW[xxyyx]   AW[xxyyy]
 - ───────── + ───────── + ───────── - ───────── - ───────── - ───────── - ───────── +
     1440         360         360         240         240         240        1440
   AW[xyxxx]   AW[xyxxy]   AW[xyxyx]   AW[xyxyy]   AW[xyyxx]   AW[xyyxy]   AW[xyyyx]
   ───────── - ───────── + ───────── + ───────── - ───────── - ───────── + ─────────}
      360         240          60         360         240         240         360
```

CW, CWSeries, tr, div

```
Deg[CW[w_]] := StringLength[w];
CWSeries[cws_Symbol][es___] := cws[es];
CWSeries[ser_Symbol][{dd_Integer}] := CWS @@ Table[ser[d], {d, dd}];
Format[s_CWSeries, StandardForm] := s[{$SeriesShowDegree}];
MakeCWSeries[cws_CWSeries] := cws;
MakeCWSeries[expr_] :=
  MakeCWSeries[expr] = MakeCWSeries[Unique[MakeCWSeries], expr];
MakeCWSeries[ser_Symbol, expr_] := (
   ser[] = Hold[MakeCWSeries[ser, expr]];
   ser[d_Integer] := ser[d] = Expand[expr /. w_CW /; Deg[w] ≠ d → 0];
   CWSeries[ser]
  );
s1_CWSeries ≡ s2_CWSeries :=
  And @@ ((s1[#] == s2[#]) & /@ Range[$SeriesCompareDegree]);
AddCWSeries[ss___CWSeries] := AddCWSeries[ss] = Module[{ser},
    ser = Unique[AddCWSeries];
    ser[] = Hold[AddCWSeries[ss]];
    ser[d_Integer] := ser[d] = Plus @@ ((#[d]) & /@ {ss});
    CWSeries[ser]
   ];
IntegrateCWSeries[cws_CWSeries, {s_, s0_, s1_}] :=
  IntegrateCWSeries[cws, {s, s0, s1}] = Module[{ser},
    ser = Unique[IntegrateCWSeries];
    ser[] = Hold[IntegrateCWSeries[cws, {s, s0, s1}]];
    ser[d_Integer] := ser[d] = Expand[Integrate[cws[d], {s, s0, s1}]];
    CWSeries[ser]
   ];
```

```
tr[w_AW] := tr[w] = CW[RotateToMinimal @@ w];
tr[expr_] := expr /. aw_AW :→ tr[aw];
tr[as_ASeries] := tr[as] = Module[{cws},
    cws = Unique[tr];
    cws[] = Hold[tr[as]];
    cws[d_] := cws[d] = tr[as[d]];
    CWSeries[cws]
   ];
```

```
tr[AW["yxxyxx"]]
```

```
CW[xxyxxy]
```

```
t1 = σ["y", BCHBase] // tr
```

```
CWSeries[tr$103]
```

```
t1[5]
```

$$\frac{\text{CW}[\text{xxxxy}]}{1440}-\frac{\text{CW}[\text{xxxyy}]}{180}+\frac{\text{CW}[\text{xxyxy}]}{120}+\frac{\text{CW}[\text{xxyyy}]}{480}-\frac{\text{CW}[\text{xyxyy}]}{720}$$

```
div[y_LW, w_LW] /; Deg[y] == 1 := div[y, w] = tr[(AW @@ y) ** σ[y, w]];
div[y_, ls_LieSeries] := div[y, ls] = Module[{cws},
     cws = Unique[div];
     cws[] = Hold[div[y, ls]];
     cws[d_] := cws[d] = div[LW[y], ls[d]];
     CWSeries[cws]
    ];
div[y_, expr_] := Expand[expr /. w_LW :> div[LW[y], w]];
```

```
{div["x", BCHBase][7], div["y", BCHBase][7]}
```

$$
\begin{aligned}
\Big\{&-\frac{\text{CW[xxxxxxy]}}{30240}+\frac{\text{CW[xxxxxyy]}}{2520}-\frac{\text{CW[xxxxyxy]}}{1008}-\frac{19\,\text{CW[xxxxyyy]}}{15120}+\frac{\text{CW[xxxyxxy]}}{2520}+\frac{\text{CW[xxxyxyy]}}{504}+\\
&\frac{\text{CW[xxxyyxy]}}{504}+\frac{19\,\text{CW[xxxyyyy]}}{15120}+\frac{\text{CW[xxyxxyy]}}{1680}-\frac{\text{CW[xxyxyxy]}}{280}-\frac{\text{CW[xxyxyyy]}}{504}-\frac{\text{CW[xxyyxyy]}}{1680}-\\
&\frac{\text{CW[xxyyyxy]}}{504}-\frac{\text{CW[xxyyyyy]}}{2520}+\frac{\text{CW[xyxyxyy]}}{280}+\frac{\text{CW[xyxyyyy]}}{1008}-\frac{\text{CW[xyyxyyy]}}{2520}+\frac{\text{CW[xyyyyyy]}}{30240},\\
&\frac{\text{CW[xxxxxxy]}}{30240}-\frac{\text{CW[xxxxxyy]}}{2520}+\frac{\text{CW[xxxxyxy]}}{1008}+\frac{19\,\text{CW[xxxxyyy]}}{15120}-\frac{\text{CW[xxxyxxy]}}{2520}-\frac{\text{CW[xxxyxyy]}}{504}-\\
&\frac{\text{CW[xxxyyxy]}}{504}-\frac{19\,\text{CW[xxxyyyy]}}{15120}-\frac{\text{CW[xxyxxyy]}}{1680}+\frac{\text{CW[xxyxyxy]}}{280}+\frac{\text{CW[xxyxyyy]}}{504}+\frac{\text{CW[xxyyxyy]}}{1680}+\\
&\frac{\text{CW[xxyyyxy]}}{504}+\frac{\text{CW[xxyyyyy]}}{2520}-\frac{\text{CW[xyxyxyy]}}{280}-\frac{\text{CW[xyxyyyy]}}{1008}+\frac{\text{CW[xyyxyyy]}}{2520}-\frac{\text{CW[xyyyyyy]}}{30240}\Big\}
\end{aligned}
$$

```
t1 = MakeCWSeries[CW["xyxyyyy"]] //
  LieDerivation[{LW["x"] → MakeLieSeries[b[LW["x"], LW["z"]]]}]
```

```
CWSeries[LieDerivationOnCWSeries$106]
```

```
t1 /@ Range[10]
```

```
{0, 0, 0, 0, 0, 0, 0, -CW[xyxyyyyz] + CW[xyxzyyyy] - CW[xyyyyxyz] + CW[xyyyyxzy], 0, 0}
```

The Meta-Cocycle J

```
J[-1, ___] = MakeCWSeries[0];
J[n_, y_LW, μ_LieSeries, s_] := J[n, y, μ, s] = Module[
    {sμ, μs},
    sμ = ScaleLieSeries[s, μ];
    μs = StableApply[LieMorphism[{y → Ad[ScaleLieSeries[-1, sμ]][LW[z]]}], μ];
    μs = μs // LieMorphism[{LW[z] → y}];
    IntegrateCWSeries[
     AddCWSeries[
      J[n - 1, y, μ, s] // LieDerivation[{y → b[μs, y]}],
      div[y, μs ]
     ],
     {s, 0, s}
    ]
   ];
J[y_LW, μ_LieSeries] := J[y, μ] = Module[{cws, s},
    cws = Unique[J];
    cws[] = Hold[J[y, μ]];
    cws[d_Integer] := cws[d] = J[d - 1, y, μ, s][d] /. s → 1;
    CWSeries[cws]
   ];
```

```
Print /@ {y0 = LW["y"], μ0 = BCHBase,
   J[0, y0, μ0, s],
   J[1, y0, μ0, s],
   J[2, y0, μ0, s],
   J[y0, μ0]
  };
```

$$\langle y\rangle$$

$$\text{LS}\left[\langle x\rangle + \langle y\rangle,\ \frac{\langle xy\rangle}{2},\ \frac{\langle xxy\rangle}{12} + \frac{\langle xyy\rangle}{12}\right]$$

$$\text{CWS}\Big[s\,\text{CW}[y],\ \frac{1}{2} s\,\text{CW}[xy] + \frac{1}{2} s^2\,\text{CW}[xy],\ \frac{1}{12} s\,\text{CW}[xxy] + \frac{1}{4} s^2\,\text{CW}[xxy] + \frac{1}{6} s^3\,\text{CW}[xxy] - \frac{1}{12} s\,\text{CW}[xyy] - \frac{1}{4} s^2\,\text{CW}[xyy] - \frac{1}{6} s^3\,\text{CW}[xyy]\Big]$$

$$\text{CWS}\Big[s\,\text{CW}[y],\ \frac{1}{2} s\,\text{CW}[xy] + \frac{1}{2} s^2\,\text{CW}[xy],\ \frac{1}{12} s\,\text{CW}[xxy] + \frac{1}{4} s^2\,\text{CW}[xxy] + \frac{1}{6} s^3\,\text{CW}[xxy] - \frac{1}{12} s\,\text{CW}[xyy] - \frac{1}{4} s^2\,\text{CW}[xyy] - \frac{1}{6} s^3\,\text{CW}[xyy]\Big]$$

$$\text{CWS}\Big[s\,\text{CW}[y],\ \frac{1}{2} s\,\text{CW}[xy] + \frac{1}{2} s^2\,\text{CW}[xy],\ \frac{1}{12} s\,\text{CW}[xxy] + \frac{1}{4} s^2\,\text{CW}[xxy] + \frac{1}{6} s^3\,\text{CW}[xxy] - \frac{1}{12} s\,\text{CW}[xyy] - \frac{1}{4} s^2\,\text{CW}[xyy] - \frac{1}{6} s^3\,\text{CW}[xyy]\Big]$$

$$\text{CWS}\left[\text{CW}[y],\ \text{CW}[xy],\ \frac{\text{CW}[xxy]}{2} - \frac{\text{CW}[xyy]}{2}\right]$$

$$\mathbf{CWS}\Big[\mathbf{s\,CW["y"],\ \frac{1}{2}\,s\,CW["xy"]+\frac{1}{2}\,s^{2}\,CW["xy"],\ \frac{1}{12}\,s\,CW["xxy"]+\frac{1}{4}\,s^{2}\,CW["xxy"]+\frac{1}{6}\,s^{3}\,CW["xxy"]-\frac{1}{12}\,s\,CW["xyy"]-\frac{1}{4}\,s^{2}\,CW["xyy"]-\frac{1}{6}\,s^{3}\,CW["xyy"]}\Big]\ \mathbf{/.\ s\to 1}$$

$$\mathrm{CWS}\Big[\mathrm{CW[y]},\ \mathrm{CW[xy]},\ \frac{\mathrm{CW[xxy]}}{2}-\frac{\mathrm{CW[xyy]}}{2}\Big]$$

```
$SeriesCompareDegree = $SeriesShowDegree = 8;
J[3, y0, μ0, s] ≡ J[4, y0, μ0, s]
```

```
True
```

```
J[y0, μ0][6]
```

$$\frac{\mathrm{CW[xxxxxy]}}{120}+\frac{31\,\mathrm{CW[xxxxyy]}}{48}-\frac{11\,\mathrm{CW[xxxyxy]}}{6}+\frac{109\,\mathrm{CW[xxxyyy]}}{36}+\frac{7\,\mathrm{CW[xxyxxy]}}{8}-\frac{23\,\mathrm{CW[xxyxyy]}}{4}-\frac{23\,\mathrm{CW[xxyyxy]}}{4}+\frac{31\,\mathrm{CW[xxyyyy]}}{48}+\frac{28\,\mathrm{CW[xyxyxy]}}{3}-\frac{11\,\mathrm{CW[xyxyyy]}}{6}+\frac{7\,\mathrm{CW[xyyxyy]}}{8}+\frac{\mathrm{CW[xyyyyy]}}{120}$$