Pensieve header: Quantum sl_n following Rosso and Yamane.

Notes.

1. Rosso is "An Analogue of P.B.W. Theorem and the Universal R-matrix for $U_\hbar(sl(N+1))$", Yamane is "A Poincare-Birkhoff-Witt Theorem for Quantized Universal Enveloping Algebras of Type A_N".
2. We call our algebra $U_q(\mathrm{sl}_{\$n})$, not $U_\hbar(\mathrm{sl}(N+1))$.
3. We dot not have a name for Rosso's X_i and Y_j. Rosso's $E_{\epsilon_i-\epsilon_{j+1}}$ is our $x[i, j] \leftrightarrow$ xij. Likewise Rosso's $\eta_{\epsilon_i-\epsilon_{j+1}}$ is our $y[i, j] \leftrightarrow$ yij.
4. Rosso's H_i is our a_i and Rosso's ξ_i is our b_i.

```
SetDirectory@"C:\\drorbn\\AcademicPensieve\\Projects\\SolvablePBW";
<< KnotTheory`
```

```
Loading KnotTheory` version of October 29, 2024, 10:29:52.1301.
Read more at http://katlas.org/wiki/KnotTheory.
```

```
δ_{ij__} := KroneckerDelta[ij];
```

```
CF[ℰ_] := If[$k == ∞,
   Expand[ℰ],
   Expand[Normal@Series[ℰ /. q → e^{ħ ϵ γ}, {ϵ, 0, $k}]]
  ];
```

```
x[i_, j_] := x <> i <> j; a[i_] := a <> i; A[i_] := A <> i;
ξ[i_, j_] := ξ <> i <> j; α[i_] := α <> i; 𝒜[i_] := 𝒜 <> i;
```

```
τ[s_Symbol] := First[Characters[SymbolName[s]]];
ι[s_Symbol] := FromDigits /@ Rest[Characters[SymbolName[s]]];
τ[s_[_]] := τ[s]; ι[s_[_]] := ι[s];
```

The following ordering agrees with Rosso's; right at the start of Section 2 and then continued implicitly within the formula for R at the start of Section 4. Simply put, each class of variables is ordered lexicographically, and the classes are ordered xAayBb.

```
(* q=e^{ħ γ ϵ}; Ai[n]=e^{ħ ϵ ai}; Bi[n]=e^{ħ γ bi} *)
```

```
$n = 3; $k = ∞; $E := {$n, $k}
```

```
$PBWGens := Flatten@{
   Table[As[i, 1, $n - 1], As[j, i + 1, $n],
    x[i, j]* = ξ[i, j]; ξ[i, j]* = x[i, j]; x[i, j]],
   Table[As[i, 1, $n - 1],
    A[i]* = 𝒜[i]; 𝒜[i]* = A[i]; A[i][_]],
   Table[As[i, 1, $n - 1],
    a[i]* = α[i]; α[i]* = a[i]; a[i]]
  }
(#_U = U_0@#) & /@ $PBWGens;
$PBWRule = Module[{k = 0}, Flatten@Table[{g → ++k, g_{i_} → {i, k}}, {g, $PBWGens}]];
x_ ≤ y_ := OrderedQ[{x, y} /. $PBWRule];
x_ < y_ := ! OrderedQ[{y, x} /. $PBWRule];
```

```
UCF[ℰ_] := Module[{F, k}, Expand@Collect[
    ℰ //. {U_i_[l___, _[0], r___] → U_i[l, r]} //. If[$k == ∞, {}, {
        U_i_[l___, AB_[n_], r___] :> Sum[As[k, 0, $k], ℏ^k ϵ^k/k! U_i[l, Sequence @@ Table[a @@ ι[AB], k], r]]
      }],
    U_[___], F
  ] /. F → CF]
```

```
m_s__[0] = 0;
m_s__[x_Plus] := UCF[m_s /@ x];
m_i_→j_[ℰ_] := ℰ /. U_i → U_j;
m_i_,j_→k_[c_. U_i_[x___] U_j_[]] := c U_k[x];
m_i_,j_→k_[c_. U_i_[] U_j_[y___]] := c U_k[y];
m_i_,j_→k_[c_. U_i_[xx___, x_[n1_]] U_j_[x_[n2_], yy___]] := UCF[c U_i[xx, x[n1 + n2]] U_j[yy]] // m_i,j→k;
m_i_,j_→k_[c_. U_i_[xx___, x_] U_j_[y_, yy___]] := If[x ≤ y,
   c U_k[xx, x, y, yy],
   ((c U_i[xx] RR_j[x, y] // UCF // m_i,j→i) U_j[yy] // UCF // m_i,j→k) // UCF
  ];
m_s__[ℰ_] := m_s[UCF[ℰ]];
```

```
Δ_i_→j_,k_[ℰ_] := UCF@Module[{j1, k1}, ℰ /. {
      U_i[] → U_j[] U_k[],
      U_i[f_, fs___] :> m_k,k1→k @ m_j,j1→j[Δ_j,k[f] Δ_i→j1,k1[U_i[fs]]]
     }];
Δ_i_→j_,k_,l__[ℰ_] := ℰ // Δ_i→j,k // Δ_k→k,l
```

```
S_i_[ℰ_] := UCF@Module[{i1}, ℰ /. {
      U_i[f_, fs___] :> m_i1,i→i[S_i[f] S_i1[U_i1[fs]]]
     }];
```

The 𝔸-algebra

```
RR_k_[ai_, aj_] /; τ[ai] == τ[aj] == "a" := U_k[aj, ai];
RR_k_[ai_, Aj_[n_]] /;  τ[ai] == "a" ∧ τ[Aj] == "A" := U_k[Aj[n], ai];
RR_k_[Ai_[m_], Aj_[n_]] /;  τ[Ai] == τ[Aj] == "A" := U_k[Aj[n], Ai[m]];
RR_k_[am_, xij_] /; τ[am] == "a" ∧ τ[xij] == "x" := Module[{i, j, m},
   {i, j} = ι@xij; {m} = ι@am;
   U_k[xij, am] + γ (δ_m,i − δ_m+1,i − δ_m,j + δ_m+1,j) U_k[xij] / 2
  ];
RR_k_[Am_[n_], xij_] /; τ[Am] == "A" ∧ τ[xij] == "x" := Module[{i, j, m},
  {i, j} = ι@xij; {m} = ι@Am; q^(-n (δ_m,i - δ_m+1,i - δ_m,j + δ_m+1,j)) U_k[xij, Am[n]]
 ]
```

We take the reduction rules RR for xij's from Yamane page 509:

(I) $i = m < j < n$, (II) $i < m < n < j$, (III) $i < m < j = n$,

(IV) $i < m < j < n$, (V) $i < j = m < n$, (VI) $i < j < m < n$.

$$
\begin{array}{ll}
q^{-2}e_{ij}e_{mn} - e_{mn}e_{ij} = 0 & \text{if } ((i,j),(m,n)) \in C_{(\mathrm{I})} \bigcup C_{(\mathrm{III})}. \\
[e_{ij}, e_{mn}] = 0 & \text{if } ((i,j),(m,n)) \in C_{(\mathrm{II})} \bigcup C_{(\mathrm{VI})}. \\
[e_{ij}, e_{mn}] = (q^2 - q^{-2})e_{in}e_{mj} & \text{if } ((i,j),(m,n)) \in C_{(\mathrm{IV})}. \\
q^2 e_{ij}e_{mn} - e_{mn}e_{ij} = q e_{in} & \text{if } ((i,j),(m,n)) \in C_{(\mathrm{V})}.
\end{array}
$$

```
RR𝓀_[xmn_, xij_] /; τ[xij] == τ[xmn] == "x" := Module[{i, j, m, n},
  {i, j} = ι@xij; {m, n} = ι@xmn;
  Which[
   (i == m ∧ j < n) ∨ (i < m < j ∧ n == j), q^-2 U𝓀[xij, xmn], (* I∨III *)
   (i < m < j ∧ n < j) ∨ (j < m), U𝓀[xij, xmn], (* II∨VI *)
   i < m < j ∧ n > j, U𝓀[xij, xmn] + (q^-2 - q^2) U𝓀[x[i, n], x[m, j]], (* IV *)
   j == m, q^2 U𝓀[xij, xmn] - q U𝓀[x[i, n]] (* V *)
  ]
 ]
```

The Co-Product, Co-Associativity, and the Hopf Axiom

```
Do[As[i, 1, $n - 1],
  Δj_,𝓀_[a[i]] = Uj[a[i]] Uk[] + Uj[] Uk[a[i]];
  Δj_,𝓀_[A[i][n_]] = Uj[A[i][n]] Uk[A[i][n]];
  Δj_,𝓀_[x[i, i + 1]] = Uj[x[i, i + 1]] Uk[A[i][1]] + Uj[A[i][-1]] Uk[x[i, i + 1]]
 ];
Do[As[i, 1, $n - 2], As[n, i + 2, $n],
 Δj_,𝓀_[x[i, n]] = mk,k1→k[mj,j1→j[
    q Δj,k[x[i, n - 1]] Δj1,k1[x[n - 1, n]] - q^-1 Δj,k[x[n - 1, n]] Δj1,k1[x[i, n - 1]]
   ]]
]
```

The Antipode

```
Do[As[i, 1, $n - 1],
  Sj_[a[i]] = -Uj[a[i]];
  Sj_[A[i][n_]] = Uj[A[i][-n]];
  Sj_[x[i, i + 1]] = -q^-2 Uj[x[i, i + 1]] (*?*)
 ];
Do[As[i, 1, $n - 2], As[n, i + 2, $n],
 Sj_[x[i, n]] = mj1,j→j[
   q Sj[x[i, n - 1]] Sj1[x[n - 1, n]] - q^-1 Sj[x[n - 1, n]] Sj1[x[i, n - 1]]
  ]
]
```

The Generating Functions for the 𝔸 operations

NonCommutativeMultiply, B, adPower, Ad, TT, Seepage

```
Supp[ℰ_] := Union@Cases[{ℰ}, U_i_[___] :→ i, ∞];
```

```
Unprotect[NonCommutativeMultiply];
NonCommutativeMultiply[x_] := x;
x_ ⊠ y_ := Module[{i, is = Supp[x] ⋂ Supp[y], σ, z},
   z = x; Do[z = m_{i→σ@i}[z], {i, is}];
   z = Expand[y z]; Do[z = m_{σ@i,i→i}[z], {i, is}]; z];
UB[x_, y_] := UCF[x ⊠ y - y ⊠ x];
```

```
(*adPower[x_,k_,Env_][c_*y_QU]:=UCF[c adPower[x,k,Env][y]];
adPower[x_,k_,Env_][y_Plus] := adPower[x,k,Env]/@ y;*)
adPower[_, 0, Env_][y_] := y;
(*adPower[x_,k_,Env_][0]=0;*)
adPower[x_, k_, Env_][y_] := (*adPower[x,k,Env][y]=*)UCF@UB[adPower[x, k - 1, Env][y], x];
adPower[x_, k_][y_] := adPower[x, k, $E][y];
```

```
Ad[0] = Identity;
```

```
Ad[c_. U_i_[xjk_] (EUs : U_[] ...)][y_] /; τ[xjk] == "x" := Module[{k = 0, s = 0, t},
  While[(t = adPower[U_i[xjk], k][y]) =!= 0,
   s += t / k!;
   ++k
  ];
  UCF[s ⊠ EUs]
 ]
```

```
Wt_aj_[x_] := Wt_aj[x] = UB[U_0[aj], U_0[x]] / U_0[x]
```

```
Ad[c_. U_i_[aj_] (EUs : U_[] ...)][y_] /; τ[aj] == "a" := (y /. U_i[gs___] :→ U_i[gs] ⅇ^(c Total[Wt_aj /@ {gs}]))
```

```
TT_λ_[] = 0; (* Translated Tangent *)
TT_λ_[xs___, x_] := UCF[Ad[x][TT_λ[xs]] + ∂_λ x];
TT_λ_[xs_List] := TT_λ @@ xs
```