RootSystems package

A subpackage for QuantumGroups v2.
Version 2.0, June 20, 2005, Scott Morrison

Introduction

Implementation

```
BeginPackage["QuantumGroups`RootSystems`", {"QuantumGroups`"}];
```

```
CartanMatrix;
CartanFactors;
LacingNumber;
Rank;
KillingForm;
ρ;
SimpleRoots;
SimpleReflection;
WeylOrbit;
RootWeightQ;
WeightsModRoots;
WeightInLatticeQ;
IntermediateLattices;
PositiveWeightQ;
InWeylPolytopeQ;
SortWeights;
SortWeightMultiplicities;
MinusculeWeightQ;
MinusculeRepresentationQ;
ReflectIntoPositiveWeylChamber;
DominantRoots;
ShortDominantRoots;
LongDominantRoots;
ShortSimpleRoots;
ShortRoots;
ShortDominantRootQ;
DualCoxeterNumber;
```

```
Begin["`Private`"];
```

```
CartanMatrix[An_] :=
 CartanMatrix[An] = Array[Switch[#1 - #2, 1, -1, 0, 2, -1, -1, _, 0] &, {n, n}]
```

```
ElementaryMatrix[n_, i0_, j0_] := Table[If[i == i0 ∧ j == j0, 1, 0], {i, 1, n}, {j, 1, n}]
```

```
CartanMatrix[Bn_] := CartanMatrix[Bn] = CartanMatrix[An] - ElementaryMatrix[n, n, n - 1]
```

```
CartanMatrix[Cn_] := CartanMatrix[Cn] = CartanMatrix[An] - ElementaryMatrix[n, n - 1, n]
```

```
CartanMatrix[Dn_] := CartanMatrix[Dn] =
  CartanMatrix[An] + ElementaryMatrix[n, n, n - 1] + ElementaryMatrix[n, n - 1, n] -
   ElementaryMatrix[n, n, n - 2] - ElementaryMatrix[n, n - 2, n]
```

$$\text{CartanMatrix}[E_6] = \begin{pmatrix} 2 & 0 & -1 & 0 & 0 & 0 \\ 0 & 2 & 0 & -1 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 \\ 0 & -1 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix};$$

$$\text{CartanMatrix}[E_7] = \begin{pmatrix} 2 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix};$$

$$\text{CartanMatrix}[E_8] = \begin{pmatrix} 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix};$$

$$\text{CartanMatrix}[F_4] = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -2 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{pmatrix};$$

$$\text{CartanMatrix}[G_2] = \begin{pmatrix} 2 & -3 \\ -1 & 2 \end{pmatrix};$$

```
CartanFactors[An_] := CartanFactors[An] = Table[1, {n}]
```

```
CartanFactors[Bn_] := CartanFactors[Bn] = Table[2, {n - 1}] ~Join~ {1}
```

```
CartanFactors[C_n_] := CartanFactors[C_n] = Table[1, {n - 1}] ~Join~ {2}
```

```
CartanFactors[D_n_] := CartanFactors[D_n] = Table[1, {n}]
```

```
CartanFactors[G_2] = {1, 3};
```

```
CartanFactors[F_4] = {2, 2, 1, 1};
```

```
CartanFactors[E_n_] := CartanFactors[E_n] = Table[1, {n}]
```

```
LacingNumber[(A | D | E)_] = 1;
LacingNumber[(B | C | F)_] = 2;
LacingNumber[G_2] = 3;
```

```
Rank[Γ_n_] := n
```

```
InverseCartanMatrix[Γ_] := InverseCartanMatrix[Γ] = Inverse[CartanMatrix[Γ]]
```

```
(*KillingForm[Γ_][x_,y_]:=Simplify[x.InverseCartanMatrix[Γ].y]*)
```

```
KillingForm[Γ_][x_, y_] := Simplify[x.Inverse[CartanFactors[Γ] × CartanMatrix[Γ]].y]
```

```
KillingForm[Γ_][x_, y_] :=
 Simplify[x.Inverse[Transpose[CartanMatrix[Γ]]].DiagonalMatrix[CartanFactors[Γ]].y]
```

```
ρ[Γ_] := Table[1, {Rank[Γ]}]
```

```
SimpleRoots[Γ_] := Transpose[CartanMatrix[Γ]]
```

```
SimpleReflection[Γ_, i_][x_] :=
 With[{λ = SimpleRoots[Γ][[i]]}, x - 2 KillingForm[Γ][λ, x] / KillingForm[Γ][λ, λ] λ]
```

```
AllSimpleReflections[Γ_][x_] := Table[SimpleReflection[Γ, i][x], {i, 1, Rank[Γ]}]
```

```
WeylOrbit[Γ_, λ_] :=
 FixedPoint[Union[Flatten[AllSimpleReflections[Γ] /@ #, 1], #] &, {λ}]
```

```
RootWeightQ[Γ_, λ_] := And @@ (IntegerQ /@
    Table[(2 KillingForm[Γ][λ, UnitVector[Rank[Γ], k]]) / (KillingForm[Γ][SimpleRoots[Γ][[k]], SimpleRoots[Γ][[k]]]), {k, 1, Rank[Γ]}])
```

```
WeightsModRoots[Γ_] := WeightsModRoots[Γ] = {ZeroVector[Rank[Γ]]}~
   Join~DeleteCases[IdentityMatrix[Rank[Γ]], λ_ /; RootWeightQ[Γ, λ]]
```

```
WeightInLatticeQ[Γ_, λ_, sublattice_] :=
 Or @@ (RootWeightQ[Γ, # + λ] & /@ WeightsModRoots[Γ][[sublattice]])
```

```
IntermediateLatticeQ[Γ_, subset_] := And @@ Flatten[
   Outer[WeightInLatticeQ[Γ, WeightsModRoots[Γ][[#1]] + WeightsModRoots[Γ][[#2]], subset] &,
    subset, subset]]
```

```
IntermediateLattices[Γ_] :=
 Cases[{1}~Join~# & /@ Subsets[Range[2, Length[WeightsModRoots[Γ]]]],
  subset_ /; IntermediateLatticeQ[Γ, subset]]
```

In PositiveWeightQ, we don't ask that the entries be integers, because Littelmann paths have vertices off the weight lattice.

```
PositiveWeightQ[Γ_][λ_] := And @@ (NonNegative /@ λ)
```

```
ReflectIntoPositiveWeylChamber[Γ_][λ_] :=
 With[{p = Position[λ, _?Negative]}, If[Length[p] == 0, λ,
   ReflectIntoPositiveWeylChamber[Γ][SimpleReflection[Γ, p[[1, 1]]][λ]]]]
```

```
WeightsOrderedQ[Γ_][λ_, μ_] :=
 And @@ (NonNegative[KillingForm[Γ][λ - μ, #]] & /@ IdentityMatrix[Rank[Γ]])
```

```
InWeylPolytopeQ[Γ_, λ_, μ_] :=
 InWeylPolytopeQ[Γ, λ, μ] = WeightsOrderedQ[Γ][λ, ReflectIntoPositiveWeylChamber[Γ][μ]]
```

```
OrderingWeight[A_n_] := OrderingWeight[A_n] = Table[1 + 10^(-5-i), {i, 1, n}]
```

```
WeightsOrderedComplete[Γ_][λ_, μ_] :=
 With[{θ = OrderingWeight[Γ]}, KillingForm[Γ][λ, θ] > KillingForm[Γ][μ, θ]]
```

```
SortWeights[Γ_][weights_] := With[{θ = OrderingWeight[Γ]},
  weights[[Ordering[-KillingForm[Γ][#, θ] & /@ (weights /. Irrep[_][λ_] :> λ)]]]]
```

```
SortWeightMultiplicities[Γ_][l_] := With[{θ = OrderingWeight[Γ]},
  l[[Ordering[-KillingForm[Γ][#[[1]], θ] & /@ (l /. Irrep[_][λ_] :→ λ)]]]]
```

```
MinusculeWeightQ[A_n_, λ_] := UnitVectorQ[λ]
```

```
MinusculeWeightQ[B_n_, λ_] := λ === UnitVector[n, n]
MinusculeWeightQ[C_n_, λ_] := λ === UnitVector[n, 1]
MinusculeWeightQ[D_n_, λ_] :=
 MemberQ[{UnitVector[n, 1], UnitVector[n, n - 1], UnitVector[n, n]}, λ]
```

```
MinusculeWeightQ[E6, {1, 0, 0, 0, 0, 0}] = True;
MinusculeWeightQ[E6, {0, 0, 0, 0, 0, 1}] = True;
```

```
MinusculeWeightQ[E7, {1, 0, 0, 0, 0, 0, 0}] = False;
MinusculeWeightQ[E7, {0, 1, 0, 0, 0, 0, 0}] = False;
MinusculeWeightQ[E7, {0, 0, 1, 0, 0, 0, 0}] = False;
MinusculeWeightQ[E7, {0, 0, 0, 1, 0, 0, 0}] = False;
MinusculeWeightQ[E7, {0, 0, 0, 0, 1, 0, 0}] = False;
MinusculeWeightQ[E7, {0, 0, 0, 0, 0, 1, 0}] = False;
MinusculeWeightQ[E7, {0, 0, 0, 0, 0, 0, 1}] = True;
```

I don't actually know whether 4 and 5 are minuscule or not. 1,2,3 and 6 aren't. You can use the following to test these:

```
With[{Γ = B4, k = 1},
 Max[Last /@ WeightMultiplicities[Γ, Irrep[Γ][UnitVector[Rank[Γ], k]]]] == 1 ∧
  WeightMultiplicity[Γ, Irrep[Γ][UnitVector[Rank[Γ], k]], ZeroVector[Rank[Γ]]] == 0]
```

```
False
```

```
With[{Γ = E7, k = 2},
 Max[Last /@ WeightMultiplicities[Γ, Irrep[Γ][UnitVector[Rank[Γ], k]]]] == 1 ∧
  WeightMultiplicity[Γ, Irrep[Γ][UnitVector[Rank[Γ], k]], ZeroVector[Rank[Γ]]] == 0]
```

```
False
```

```
MinusculeWeightQ[_, _] := False
```

```
MinusculeRepresentationQ[Γ_, Irrep[Γ_][λ_]] := MinusculeWeightQ[Γ, λ]
```

```
MinusculeRepresentations[Γ_] :=
 Irrep[Γ] /@ Select[IdentityMatrix[Rank[Γ]], MinusculeWeightQ[Γ, #] &]
```

```
DominantRoots[Γ_] :=
 DominantRoots[Γ] = Union[ReflectIntoPositiveWeylChamber[Γ] /@ SimpleRoots[Γ]]
```

```
ShortDominantRoots[Γ_] :=
 ShortDominantRoots[Γ] = Module[{dr = DominantRoots[Γ], shortLength},
   shortLength = Min[KillingForm[Γ][#, #] & /@ dr];
   Select[dr, KillingForm[Γ][#, #] == shortLength &]
  ]
```

```
LongDominantRoots[Γ_] :=
 LongDominantRoots[Γ] = Module[{dr = DominantRoots[Γ], longLength},
   longLength = Max[KillingForm[Γ][#, #] & /@ dr];
   Select[dr, KillingForm[Γ][#, #] == longLength &]
  ]
```

```
ShortSimpleRoots[Γ_] := ShortSimpleRoots[Γ] = Module[{shortLength},
   shortLength = Min[KillingForm[Γ][#, #] & /@ SimpleRoots[Γ]];
   Select[SimpleRoots[Γ], KillingForm[Γ][#, #] == shortLength &]
  ]
```

```
ShortRoots[Γ_] :=
 ShortRoots[Γ] = ShortRoots[Γ] = Union @@ (WeylOrbit[Γ, #] & /@ ShortDominantRoots[Γ])
```

```
ShortDominantRootQ[Γ_, λ_] := MemberQ[ShortDominantRoots[Γ], λ]
```

```
DualCoxeterNumber[A_n_] := n + 1
DualCoxeterNumber[B_n_] := 2 n - 1
DualCoxeterNumber[C_n_] := n + 1
DualCoxeterNumber[D_n_] := 2 n - 2
DualCoxeterNumber[E_6] = 12;
DualCoxeterNumber[E_7] = 18;
DualCoxeterNumber[E_8] = 30;
DualCoxeterNumber[F_4] = 9;
DualCoxeterNumber[G_2] = 4;
```

```
End[];
```

```
EndPackage[];
```

```
CartanMatrix[G_2]
```

{{2, -3}, {-1, 2}}

```
CartanFactors[G_2]
```

{1, 3}

```
CartanFactors[G_2] × CartanMatrix[G_2] // MatrixForm
```

$$\begin{pmatrix} 2 & -3 \\ -3 & 6 \end{pmatrix}$$

```
ShortRoots[F4]
```

```
{{-2, 1, 0, 0}, {-1, -1, 1, 0}, {-1, 0, 0, 0}, {-1, 0, 0, 1}, {-1, 0, 1, -1}, {-1, 1, -1, 1},
 {-1, 1, 0, -1}, {-1, 1, 0, 0}, {-1, 2, -1, 0}, {0, -1, 0, 1}, {0, -1, 1, -1}, {0, -1, 1, 0},
 {0, 1, -1, 0}, {0, 1, -1, 1}, {0, 1, 0, -1}, {1, -2, 1, 0}, {1, -1, 0, 0}, {1, -1, 0, 1},
 {1, -1, 1, -1}, {1, 0, -1, 1}, {1, 0, 0, -1}, {1, 0, 0, 0}, {1, 1, -1, 0}, {2, -1, 0, 0}}
```

```
DominantRoots[A4]
```

```
{{1, 0, 0, 1}}
```

```
ShortDominantRoots[B6]
```

```
{{0, 1, 0, 0, 0, 0}}
```

```
DominantRoots[B6]
```

```
{{0, 1, 0, 0, 0, 0}, {2, 0, 0, 0, 0, 0}}
```

```
DominantRoots[C7]
```

```
{{0, 1, 0, 0, 0, 0, 0}, {1, 0, 0, 0, 0, 0, 0}}
```

```
DominantRoots[D8]
```

```
{{0, 1, 0, 0, 0, 0, 0, 0}}
```

```
DominantRoots[G2]
```

```
{{0, 1}, {1, 0}}
```

```
ShortDominantRoots[G2]
```

```
{{0, 1}}
```

```
ReflectIntoPositiveWeylChamber[B3] /@ CartanMatrix[B3]
```

```
{{0, 1, 0}, {0, 1, 0}, {2, 0, 0}}
```

```
WeylOrbit[B3, #] & /@ Table[SimpleRoot[B3, i], {i, 1, 3}]
```

```
{{{-2, 1, 0}, {-1, -1, 1}, {-1, 0, 1}, {-1, 1, -1}, {-1, 2, -1}, {0, -1, 0},
  {0, 1, 0}, {1, -2, 1}, {1, -1, 1}, {1, 0, -1}, {1, 1, -1}, {2, -1, 0}},
 {{-2, 1, 0}, {-1, -1, 1}, {-1, 0, 1}, {-1, 1, -1}, {-1, 2, -1}, {0, -1, 0},
  {0, 1, 0}, {1, -2, 1}, {1, -1, 1}, {1, 0, -1}, {1, 1, -1}, {2, -1, 0}},
 {{-2, 0, 0}, {-2, 2, 0}, {0, -2, 2}, {0, 2, -2}, {2, -2, 0}, {2, 0, 0}}}
```