Banana Seifert

In[•]:=
```
SetDirectory["C:\\Users\\T15Roland\\Wiskunde\\Bn\\Seifert"]
Once[<< KnotTheory`]
```

Out[•]=
```
C:\Users\T15Roland\Wiskunde\Bn\Seifert
```

ParentDirectory: Argument File should be a positive machine–size integer, a nonempty string, or a File specification.

ParentDirectory: Argument File should be a positive machine–size integer, a nonempty string, or a File specification.

ToFileName: String or list of strings expected at position 1 in ToFileName[{File, WikiLink, mathematica}].

ToFileName: String or list of strings expected at position 1 in ToFileName[{File, QuantumGroups}].

```
Loading KnotTheory` version of September 6, 2014, 13:37:37.2841.
Read more at http://katlas.org/wiki/KnotTheory.
```

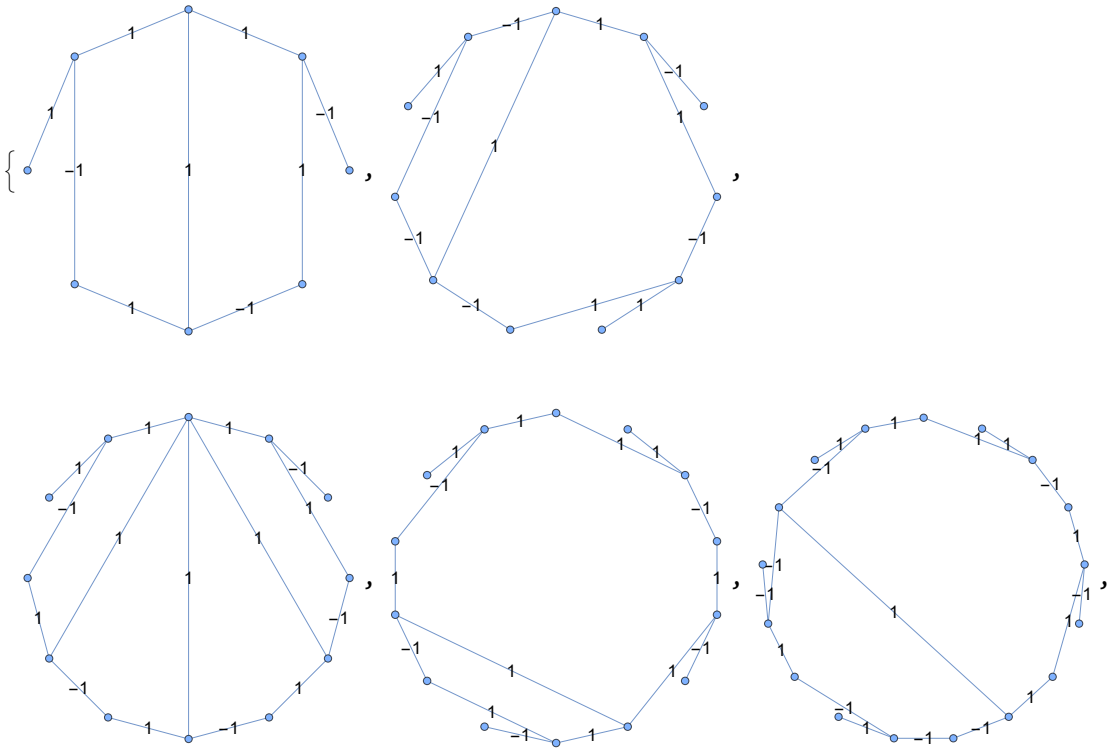
SetDelayed: Tag Diff in Diff[K_PD, rut_, ag_, n_, m_] is Protected.

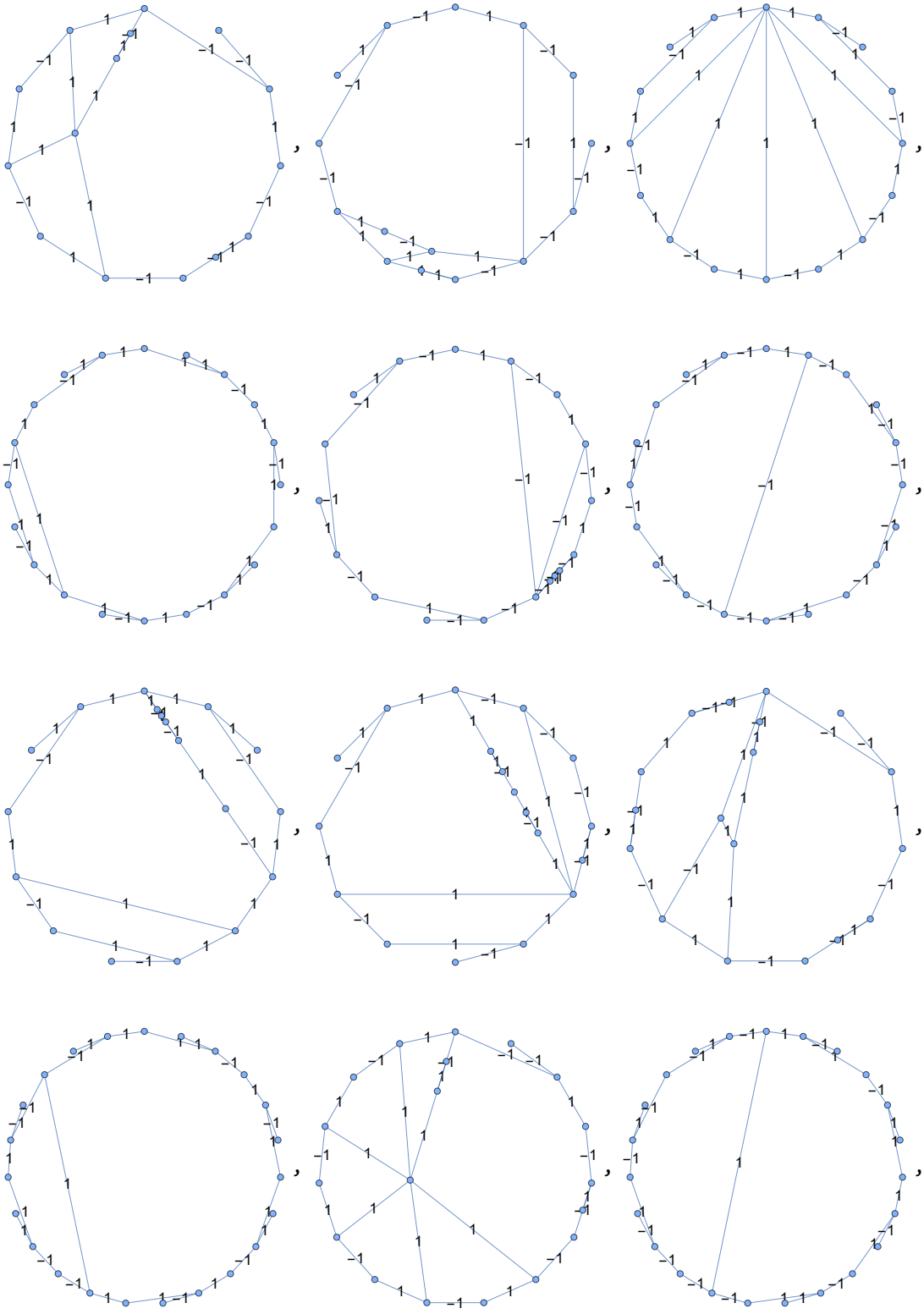
In[•]:=
```
SetAttributes[P, Orderless]
ProcessSigns[pd_] :=
 pd /. X[a_, b_, c_, d_] :> If[PositiveQ[X[a, b, c, d]], Xp[a, b, c, d], Xm[a, b, c, d]]
SeifertCycles[pd_] := (Times @@ pd) /.
    {Xp[a_, b_, c_, d_] :> P[a, b] P[c, d], Xm[a_, b_, c_, d_] :> P[a, d] P[c, b]} //.
  {P[a___, b_] P[b_, c___] :> P[a, b, c], x_^2 -> x}
SeifertBycles[pd_] := (Times @@ pd) /.
    {Xp[a_, b_, c_, d_] :> Q[a] Q[b] Q[d, c], Xm[a_, b_, c_, d_] :> Q[a, d] Q[c] Q[b]} //.
  {Q[a___, b_] Q[b_, c___] :> Q[a, b, c]}
CutList[pd_] := First /@ List @@ (SeifertCycles[pd])
IsIncoming_z_[XX_] :=
 If[(z == 1) || (Head[XX] === Xp && (z == 4)) || (Head[XX] === Xm && (z == 2)), True, False]
CutPD[pd_] := Module[{pd2 = pd, pos},
  Do[
   pos = Position[pd, c];
   If[IsIncoming_pos[[1,2]][pd[[pos[[1, 1]]]]],
    Part[pd2, Sequence @@ pos[[1]]] = $[c];
    Part[pd2, Sequence @@ pos[[2]]] = $$[c],
    Part[pd2, Sequence @@ pos[[2]]] = $[c];
    Part[pd2, Sequence @@ pos[[1]]] = $$[c]];

   , {c, CutList[pd]}];
  pd2]
SeifertBGraph[K_] :=
 Module[{pd2 = CutPD[ProcessSigns@PD[K]], Edges, Signs = <||>, XV, OV, V},
  XV =
   (List @@ pd2) /. {Xp[a_, b_, c_, d_] :> {-a, b, -d}, Xm[a_, b_, c_, d_] :> {-a, -b, c}};
```

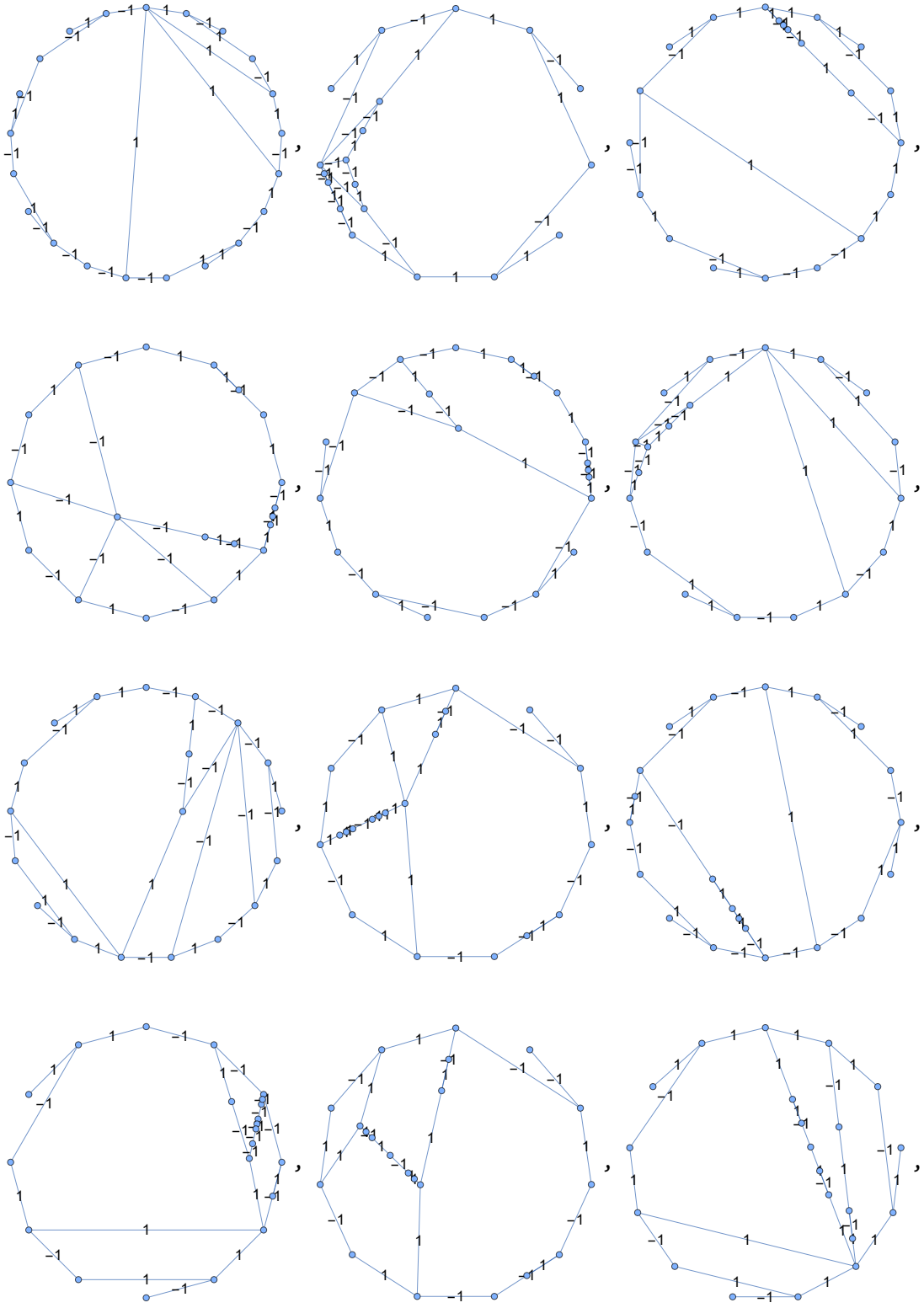
```
  Edges = XV // Flatten;
  (List @@ pd2) /. {Xp[a_, b_, c_, d_] :> (Signs = Append[Signs, {-a -> -1, b -> 1, -d -> -1}]),
    Xm[a_, b_, c_, d_] :> (Signs = Append[Signs, {-a -> 1, c -> 1, -b -> -1}])};
  OV = If[Length[#] == 1, Switch[Head[#[[1]]], Integer, {-#[[1]], #[[1]]}, $, -#, $$, #],
      Join[If[IntegerQ[#[[1]]], {#[[1]]}, {}],
       -Most[#],
       If[IntegerQ[#[[-1]]], {-#[[-1]]}, {}]
      ]

     ] & /@ (List @@ (SeifertBycles[pd2] /. x_^2 -> x) /. Q -> List);
  {V = Join[OV, XV], Signs}
 ]

(*Warning! misrepresents the labels of multi-edges*)
DrawGraph[sb_] :=
 PlanarGraph[Table[Labeled[
    UndirectedEdge @@ (First /@ Position[sb[[1]], e]), sb[[2]][e]], {e, sb[[2]] // Keys}]
 ]
```

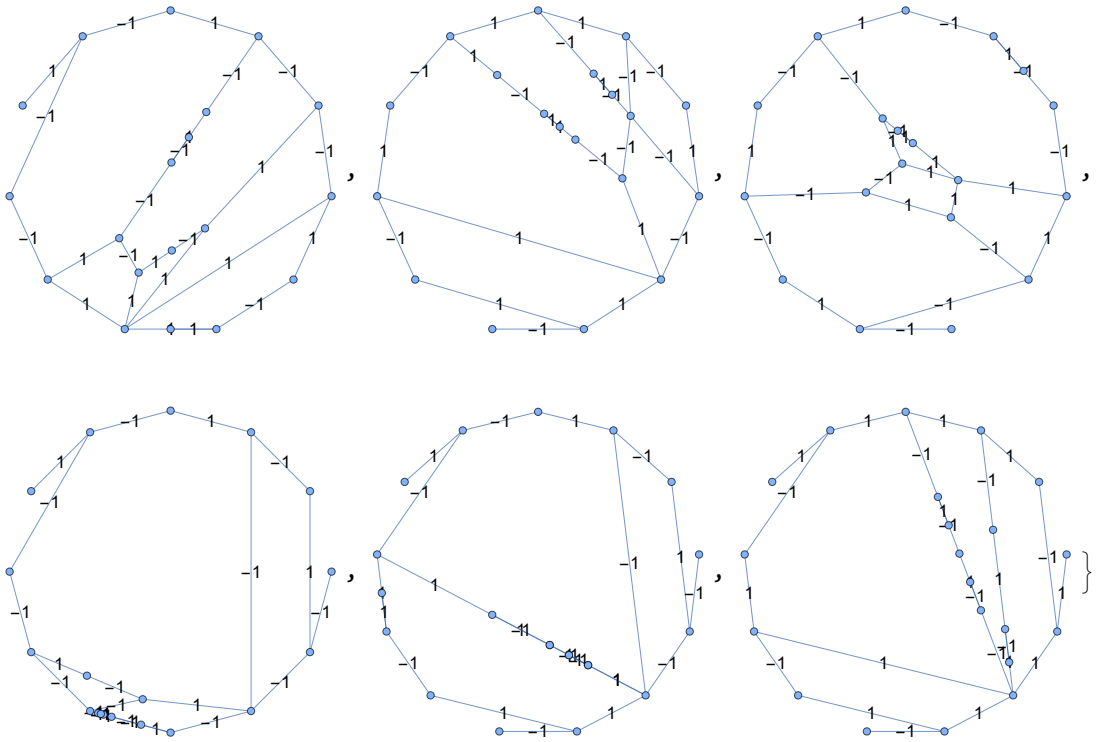
In[•]:=
```
DrawGraph[SeifertBGraph[#]] & /@ AllKnots[{3, 8}]
```

Out[•]=



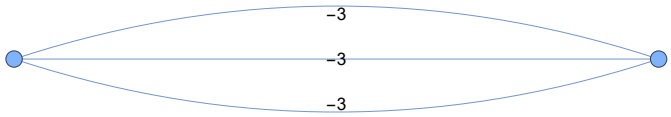






In[•]:=
```
(*Get rid of leaves and degree 2 vertices*)
Prune[sb_] := Module[{sbnew = sb, Leaves = Flatten@Select[sb[[1]], Length[#] == 1 &]},
  Do[
   sbnew[[1]] = DeleteCases[sbnew[[1]], lv, {2}];
   sbnew[[2]] = Delete[sbnew[[2]], Key[lv]]
   , {lv, Leaves}];
  sbnew /. {{} → Nothing}]
Contract[sb_] :=
 Module[{sbnew = sb, SmallV = First@Select[sb[[1]], Length[#] == 2 &], asso},
  asso = sbnew[[2]];
  asso [First@SmallV] = asso[First@SmallV] + asso[Last@SmallV];
  sbnew[[2]] = Delete[asso, Key[Last@SmallV]];

  sbnew[[1]] = DeleteCases[sbnew[[1]], SmallV];
  sbnew[[1]] = sbnew[[1]] /. {Last@SmallV → First@SmallV};
  sbnew]
```

```
In[•]:= (*Not correct!*)
        DrawGraph@{{{6, -$[2], -4}, {-$[2], 6, -4}}, <|-4 → -3, 6 → 1, -$[2] → 1|>}
Out[•]=
```

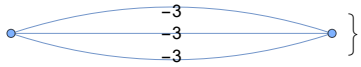


Seifert surfaces for the Rolfsen table (warning twists on multi-edges are displayed incorrectly!)

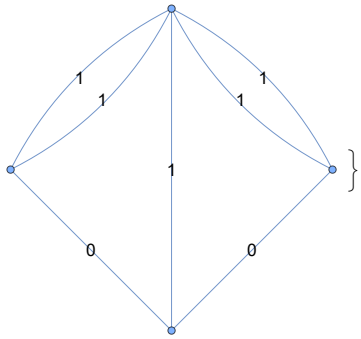
```
In[•]:= {#, (DrawGraph@Quiet@FixedPoint[Contract, Prune[SeifertBGraph[#]]])} & /@
          AllKnots[{3, 10}] // Column
Out[•]=
```

Knot[3, 1],

Knot[4, 1],

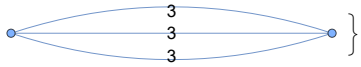


Knot[5, 1],

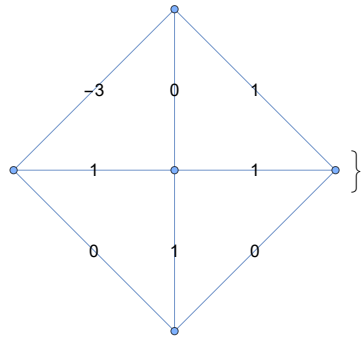


Knot[5, 2],

Knot[6, 1],

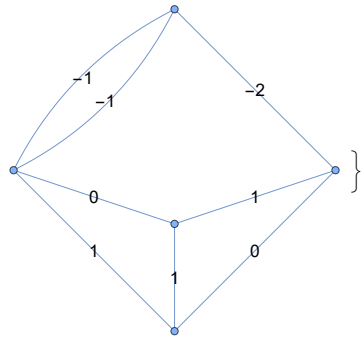


{Knot[6, 2],



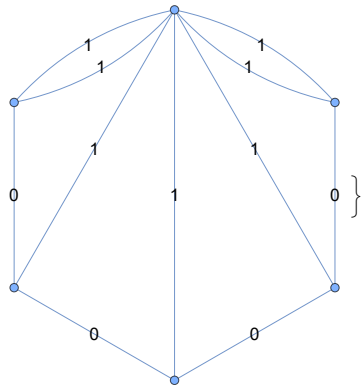
}

{Knot[6, 3],



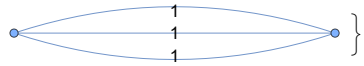
}

{Knot[7, 1],



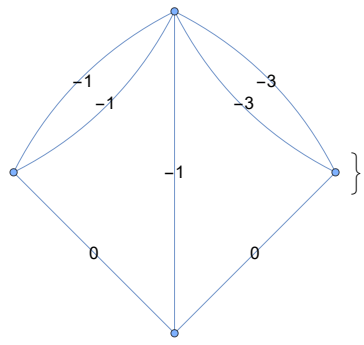
}

{Knot[7, 2],



}

{Knot[7, 3],

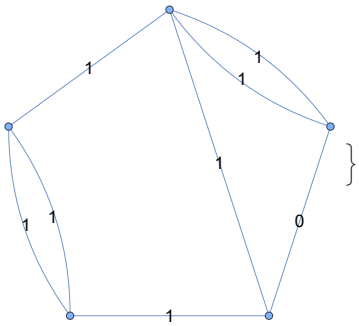


}

{Knot[7, 4],

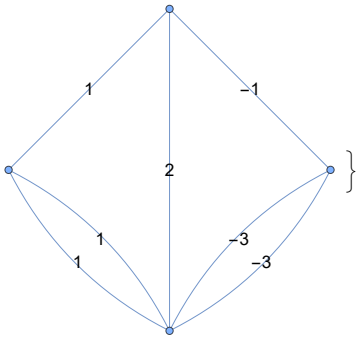
}

{Knot[7, 5],



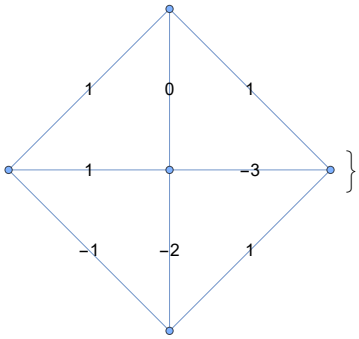
}

{Knot[7, 6],



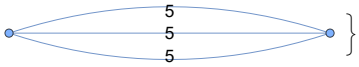
}

{Knot[7, 7],



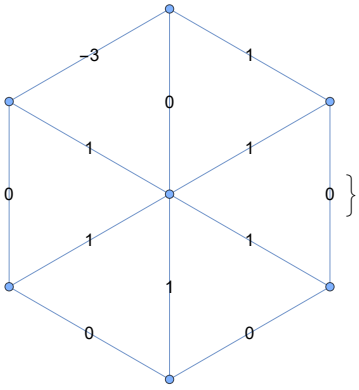
}

{Knot[8, 1],

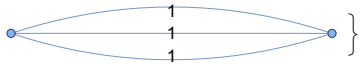


}

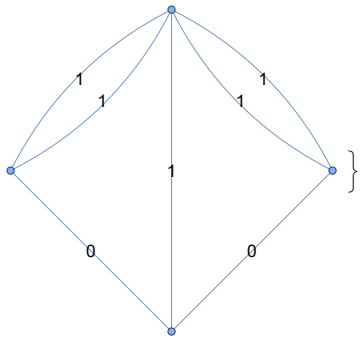
{Knot[8, 2],



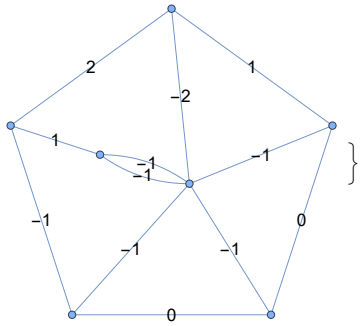
}

{Knot[8, 3], 

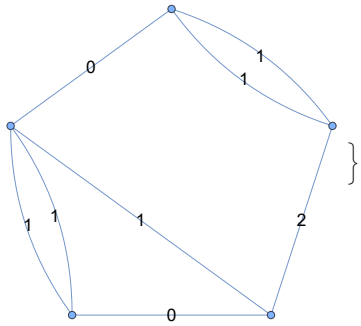
}

{Knot[8, 4], 

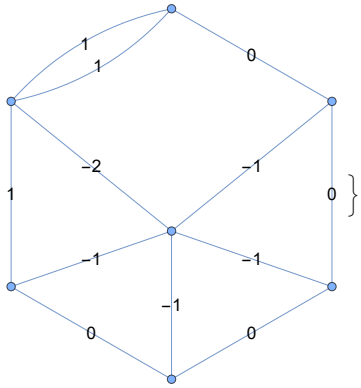
}

{Knot[8, 5], 

}

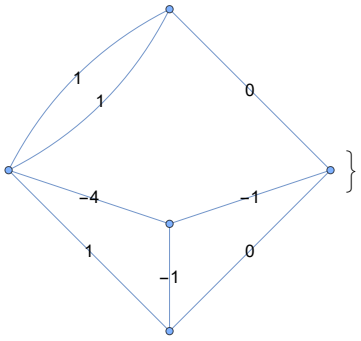
{Knot[8, 6], 

}

{Knot[8, 7], 

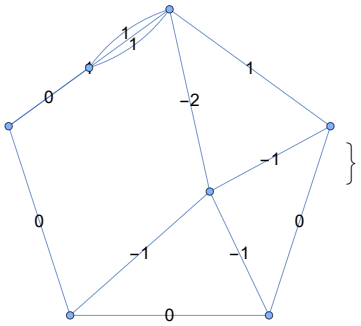
}

{Knot[8, 8],



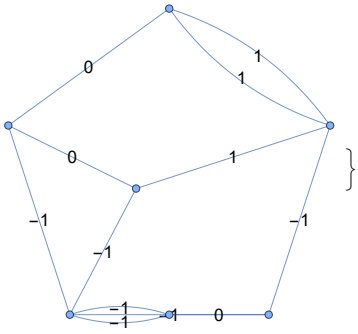
}

{Knot[8, 9],



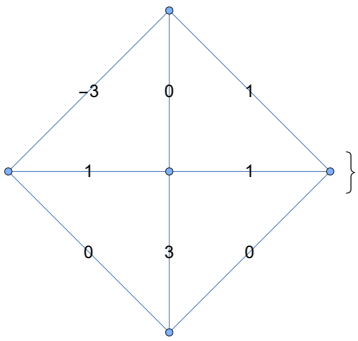
}

{Knot[8, 10],



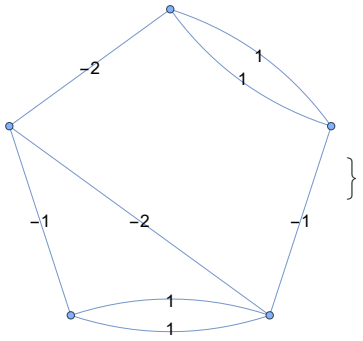
}

{Knot[8, 11],



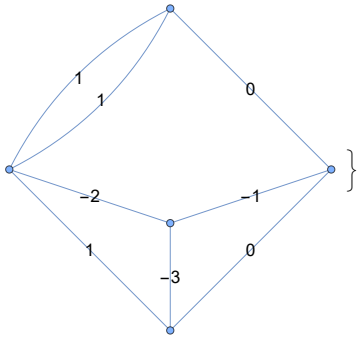
}

{Knot[8, 12],



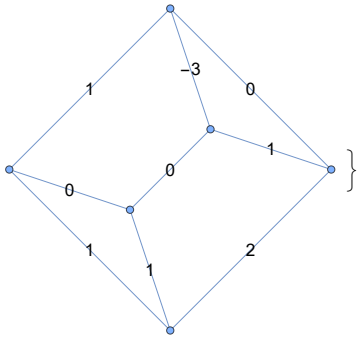
}

{Knot[8, 13],



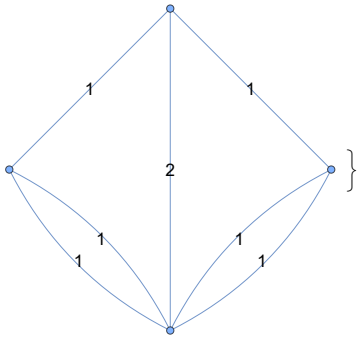
}

{Knot[8, 14],



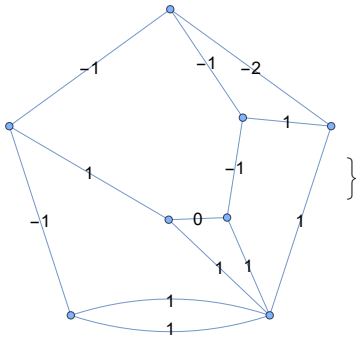
}

{Knot[8, 15],



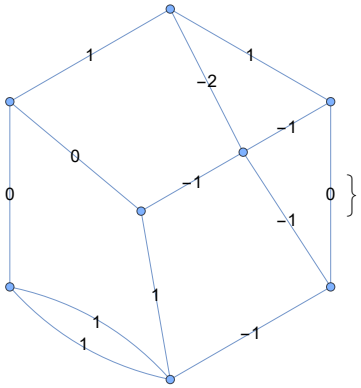
}

{Knot[8, 16],



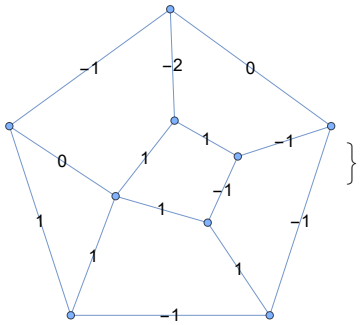
}

{Knot[8, 17],



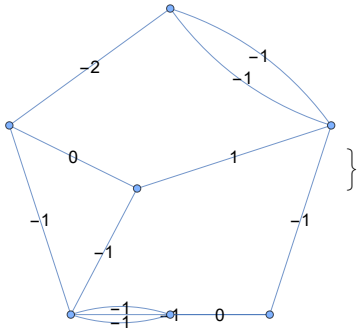
}

{Knot[8, 18],



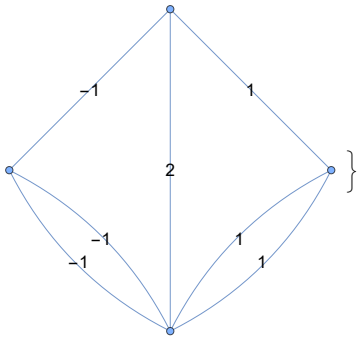
}

{Knot[8, 19],



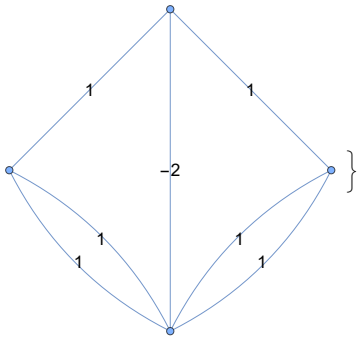
}

{Knot[8, 20],



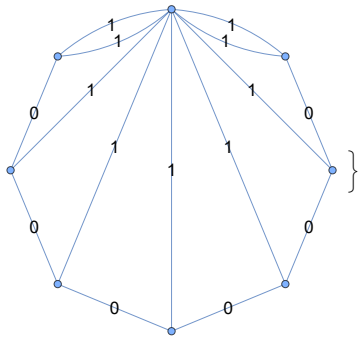
}

{Knot[8, 21],



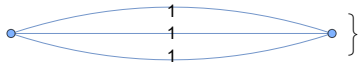
}

{Knot[9, 1],



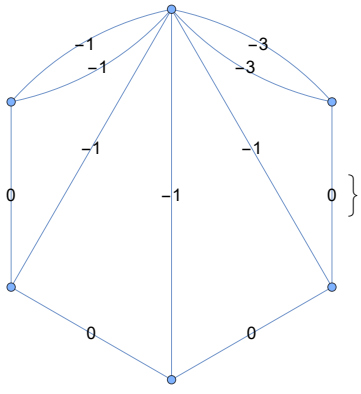
}

{Knot[9, 2],



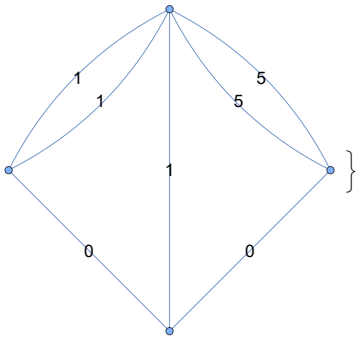
}

{Knot[9, 3],



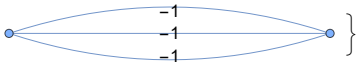
}

{Knot[9, 4],



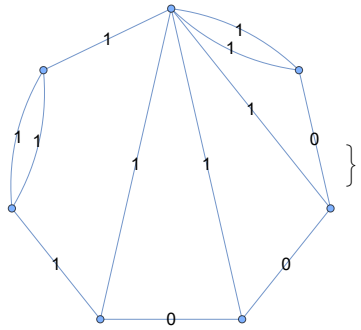
}

{Knot[9, 5],



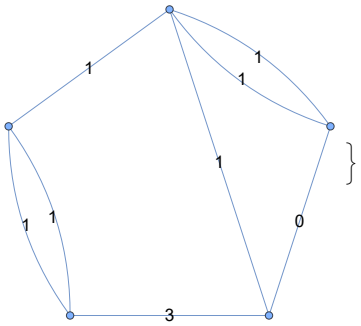
}

{Knot[9, 6],



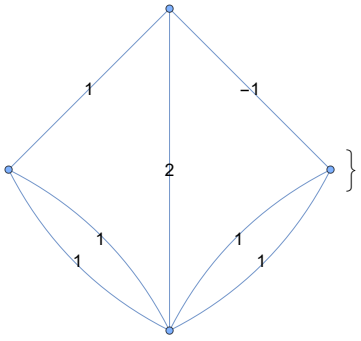
}

{Knot[9, 7],



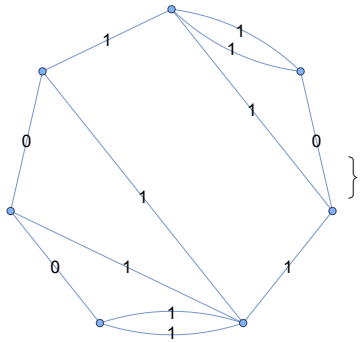
}

{Knot[9, 8],



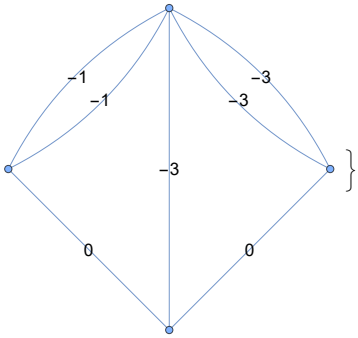
}

{Knot[9, 9],



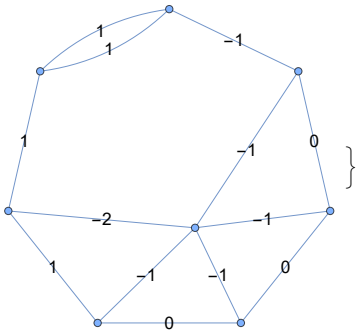
}

{Knot[9, 10],



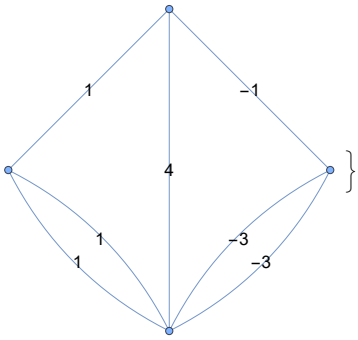
}

{Knot[9, 11],



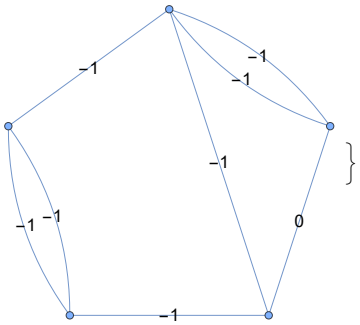
}

{Knot[9, 12],



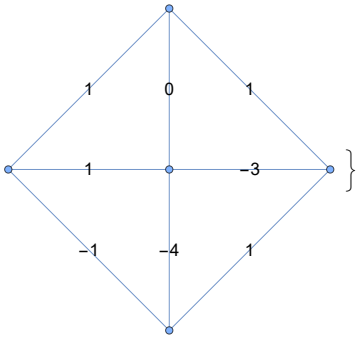
}

{Knot[9, 13],



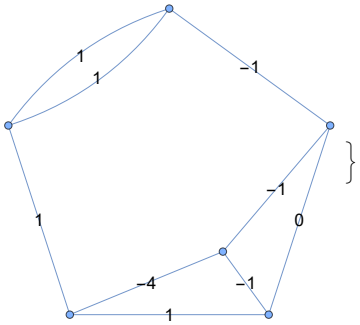
}

{Knot[9, 14],



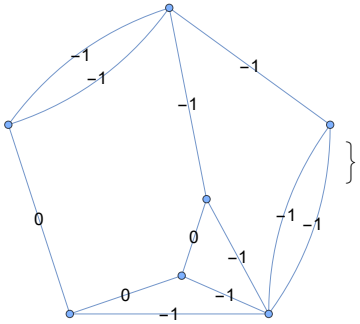
}

{Knot[9, 15],



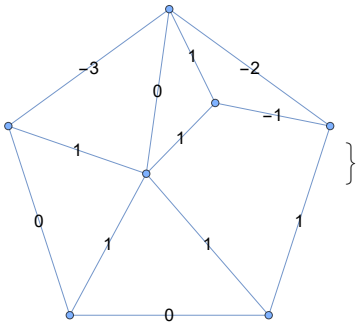
}

{Knot[9, 16],



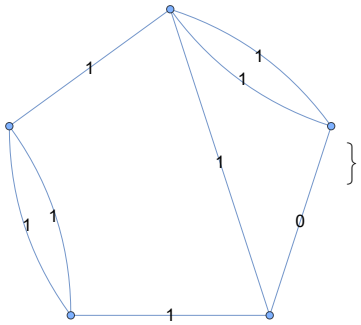
}

{Knot[9, 17],



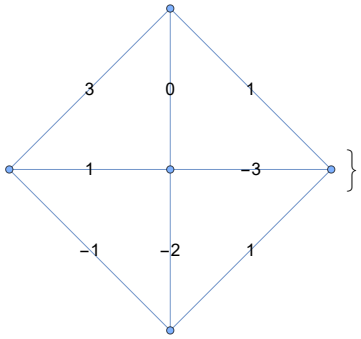
}

{Knot[9, 18],



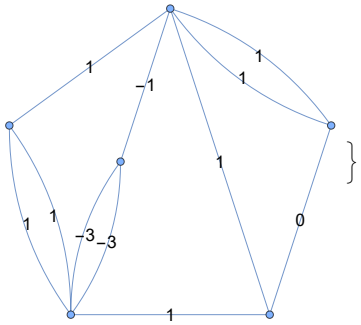
}

{Knot[9, 19],



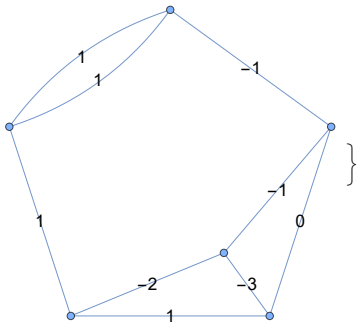
}

{Knot[9, 20],



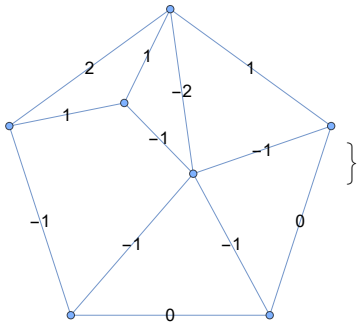
}

{Knot[9, 21],



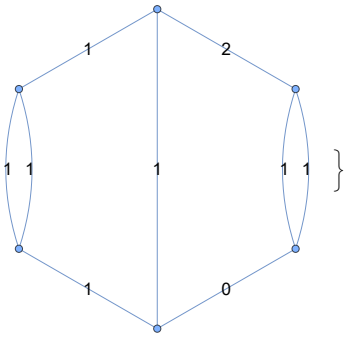
}

{Knot[9, 22],



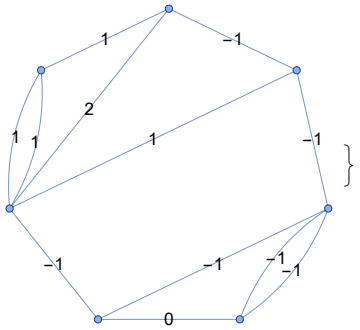
}

{Knot[9, 23],



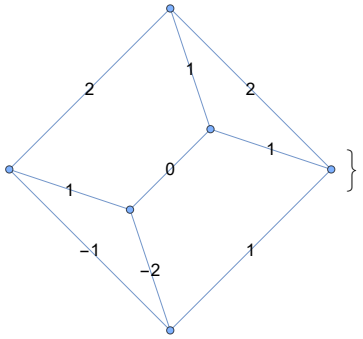
}

{Knot[9, 24],



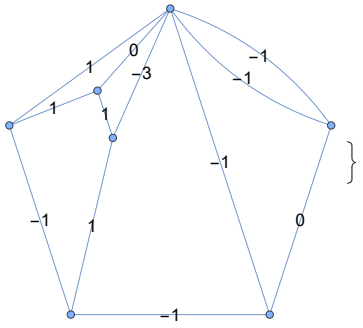
}

{Knot[9, 25],



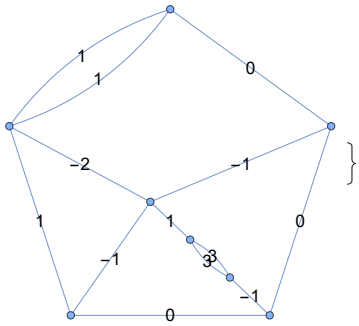
}

{Knot[9, 26],



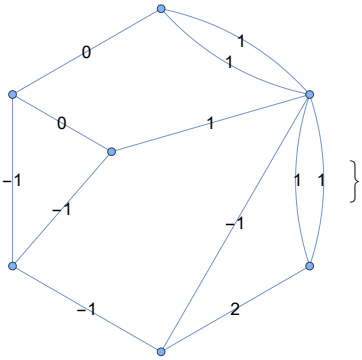
}

{Knot[9, 27],



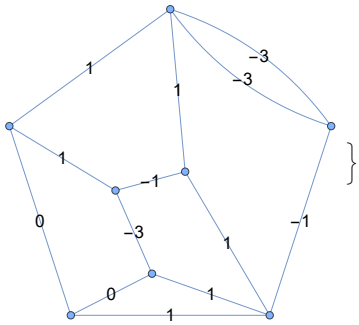
}

{Knot[9, 28],



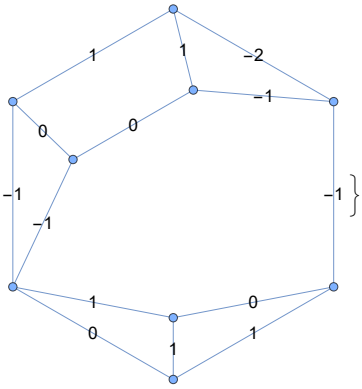
}

{Knot[9, 29],

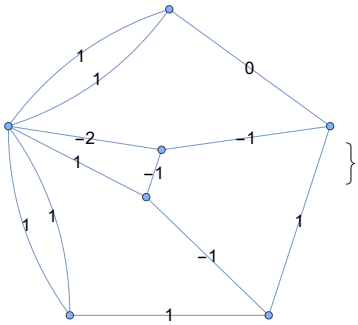


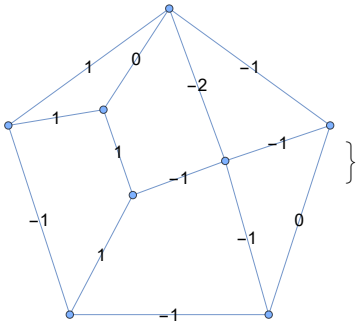
}

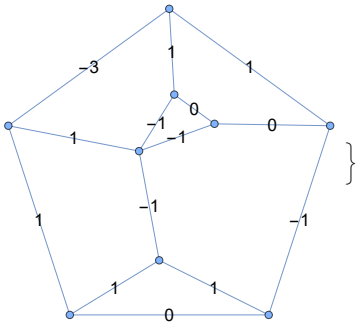
{Knot[9, 30],

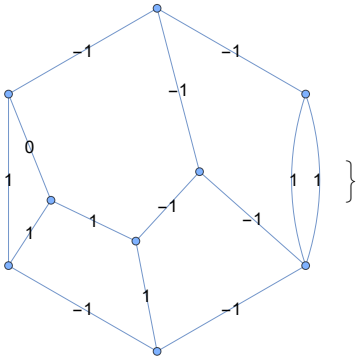


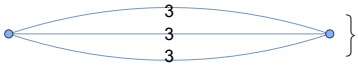
}

{Knot[9, 31],  }

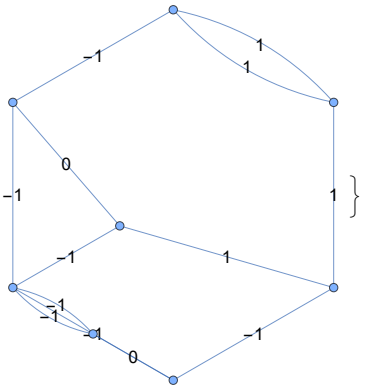
{Knot[9, 32],  }

{Knot[9, 33],  }

{Knot[9, 34],  }

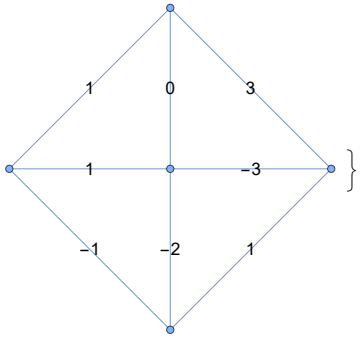
{Knot[9, 35],  }

{Knot[9, 36],



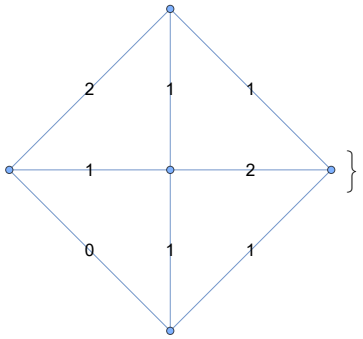
}

{Knot[9, 37],



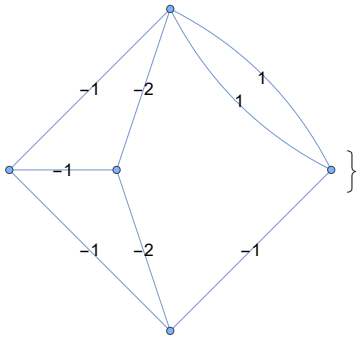
}

{Knot[9, 38],



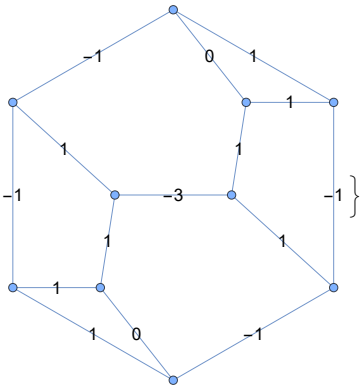
}

{Knot[9, 39],



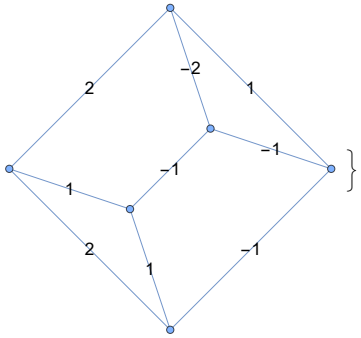
}

{Knot[9, 40],



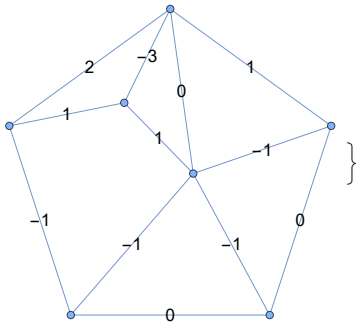
}

{Knot[9, 41],



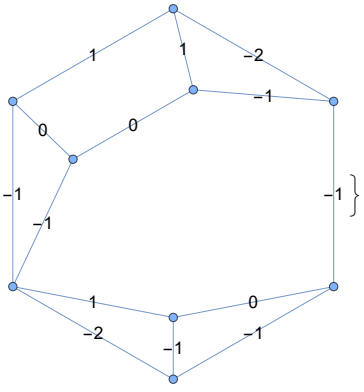
}

{Knot[9, 42],

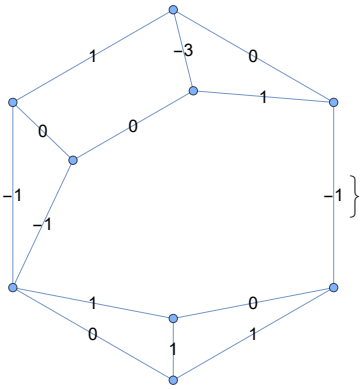


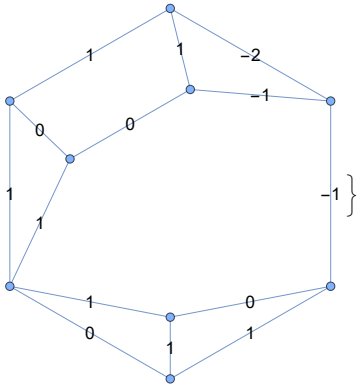
}

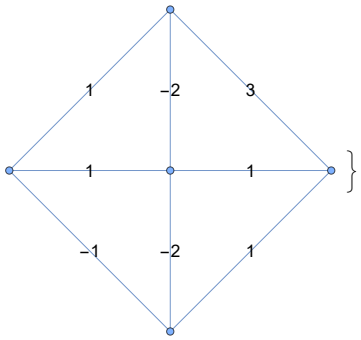
{Knot[9, 43],

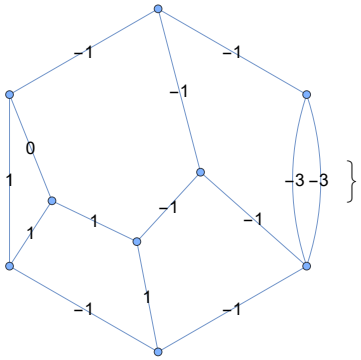


}

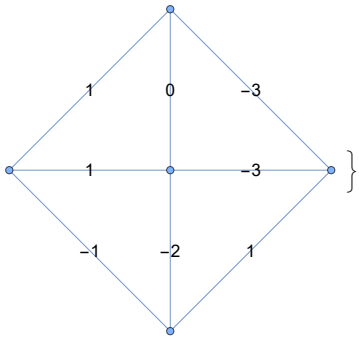
{Knot[9, 44], 

{Knot[9, 45], 

{Knot[9, 46], 

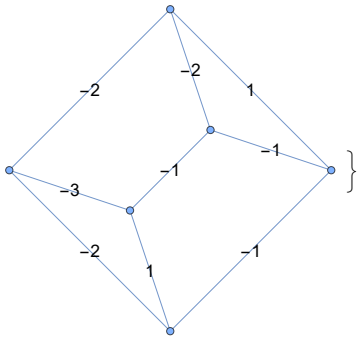
{Knot[9, 47], 

{Knot[9, 48],



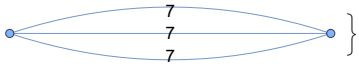
}

{Knot[9, 49],



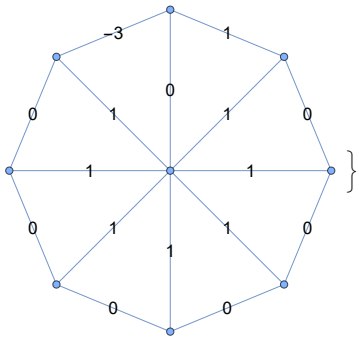
}

{Knot[10, 1],



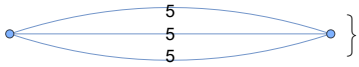
}

{Knot[10, 2],



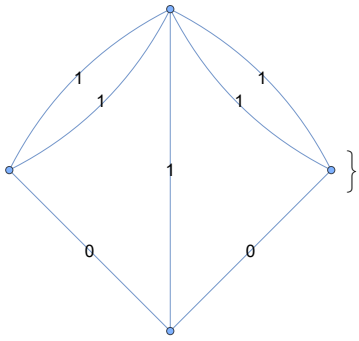
}

{Knot[10, 3],



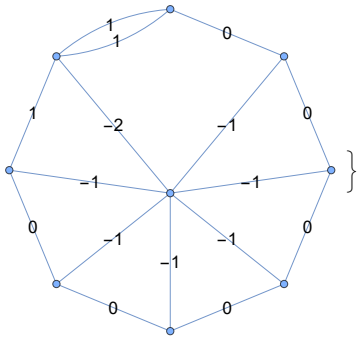
}

{Knot[10, 4],



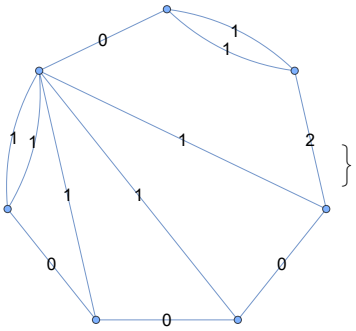
}

{Knot[10, 5],



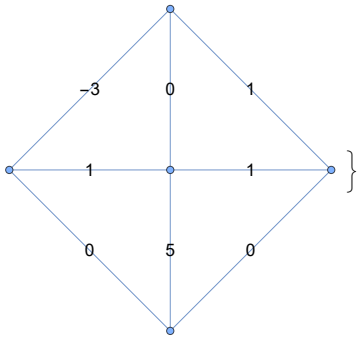
}

{Knot[10, 6],



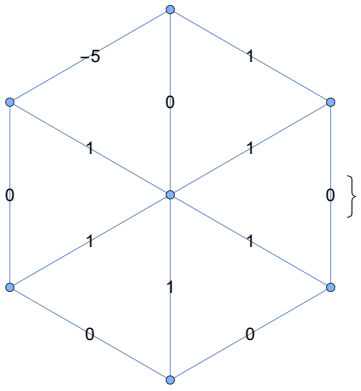
}

{Knot[10, 7],



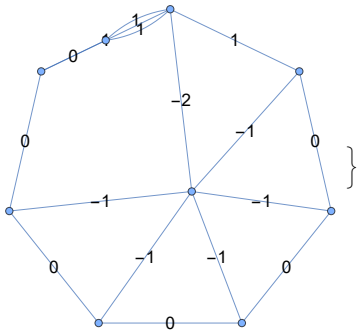
}

{Knot[10, 8],



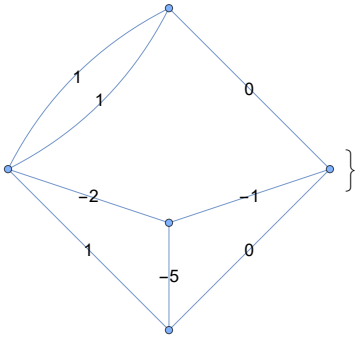
}

{Knot[10, 9],



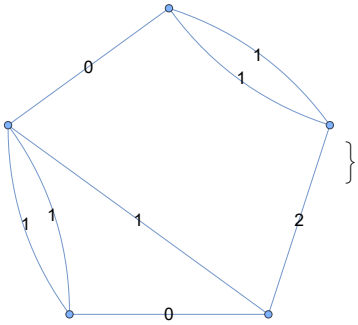
}

{Knot[10, 10],



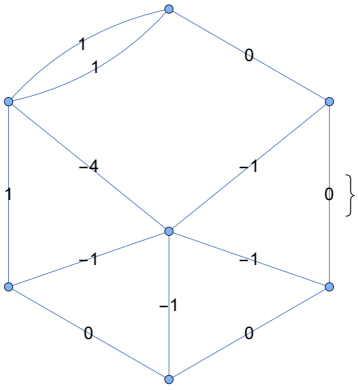
}

{Knot[10, 11],



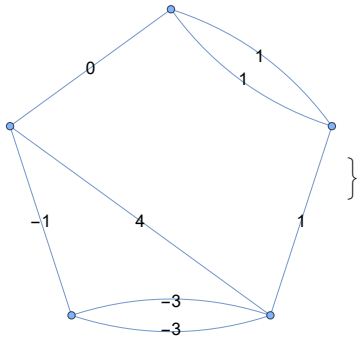
}

{Knot[10, 12],



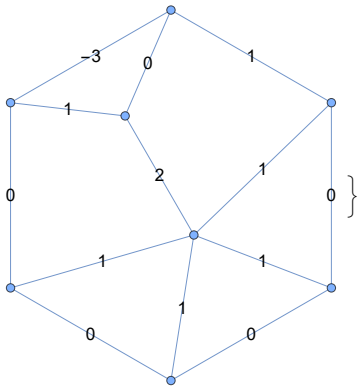
}

{Knot[10, 13],



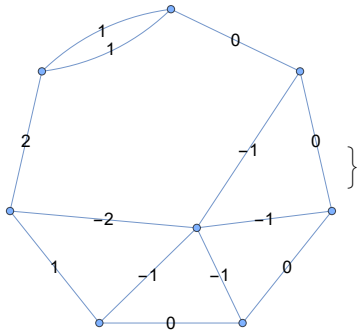
}

{Knot[10, 14],



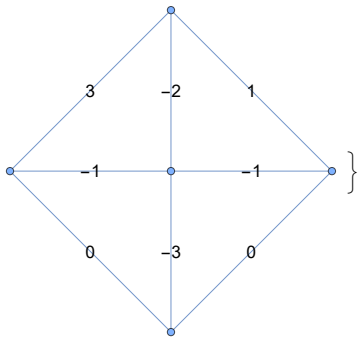
}

{Knot[10, 15],



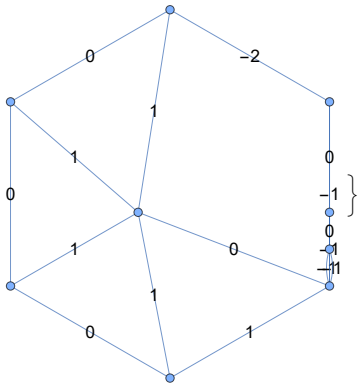
}

{Knot[10, 16],



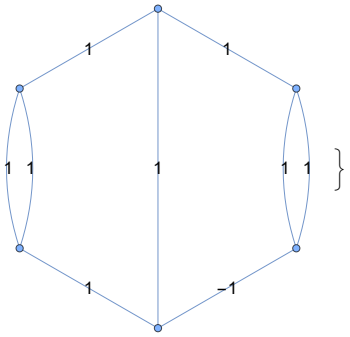
}

{Knot[10, 17],



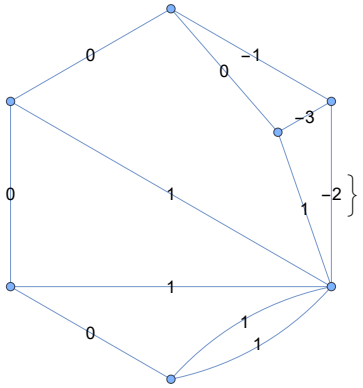
}

{Knot[10, 18],



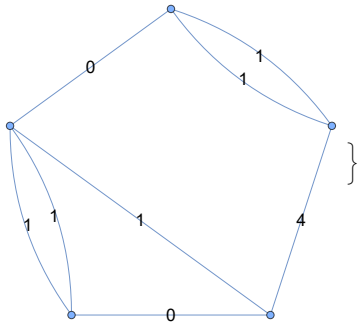
}

{Knot[10, 19],



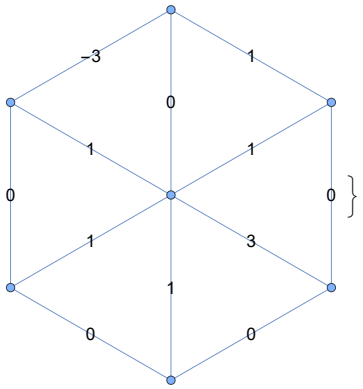
}

{Knot[10, 20],



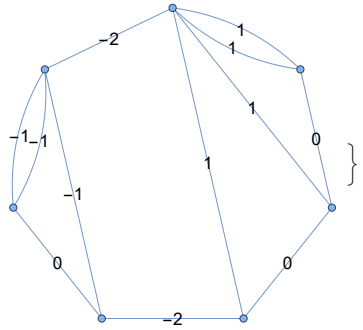
}

{Knot[10, 21],



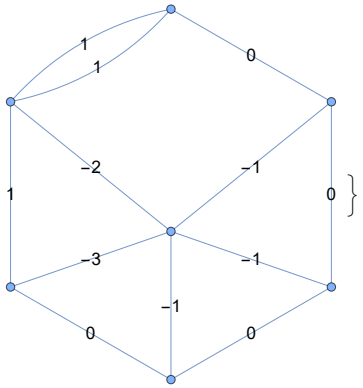
}

{Knot[10, 22],



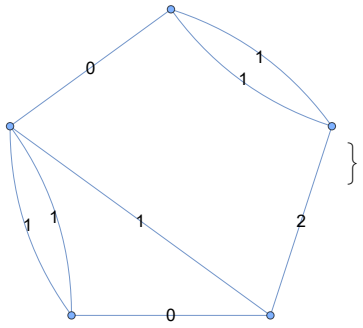
}

{Knot[10, 23],



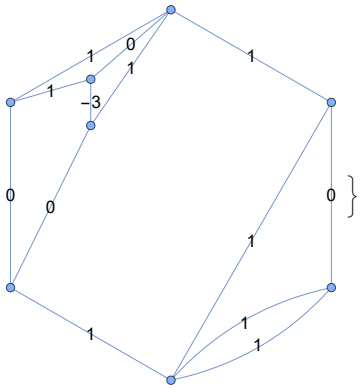
}

{Knot[10, 24],



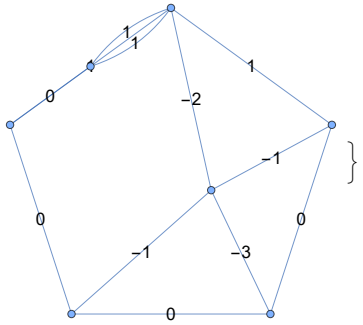
}

{Knot[10, 25],



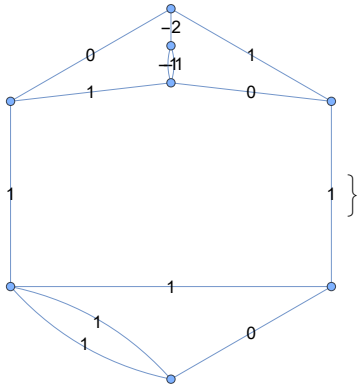
}

{Knot[10, 26],



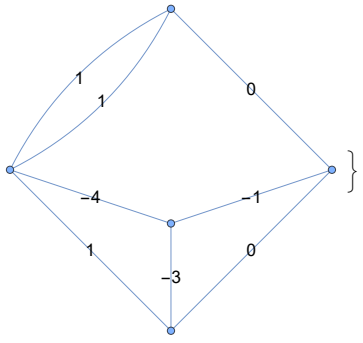
}

{Knot[10, 27],



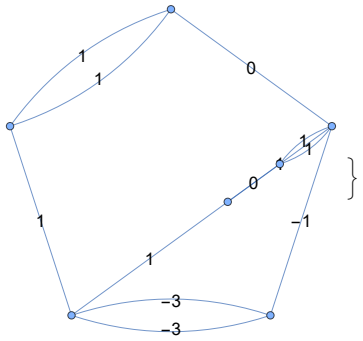
}

{Knot[10, 28],



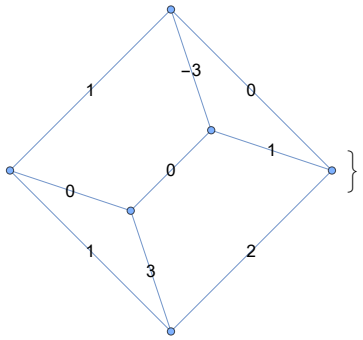
}

{Knot[10, 29],



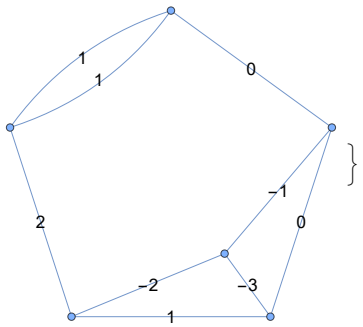
}

{Knot[10, 30],



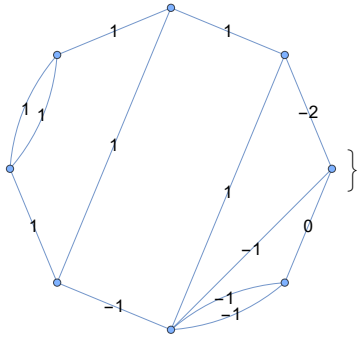
}

{Knot[10, 31],



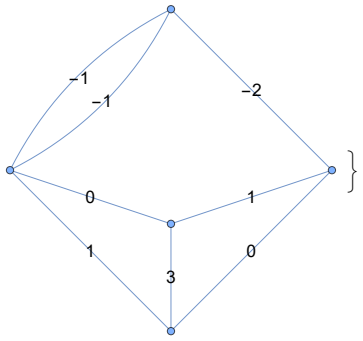
}

{Knot[10, 32],



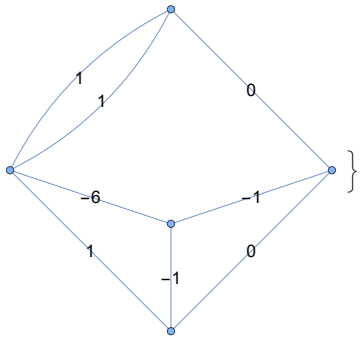
}

{Knot[10, 33],



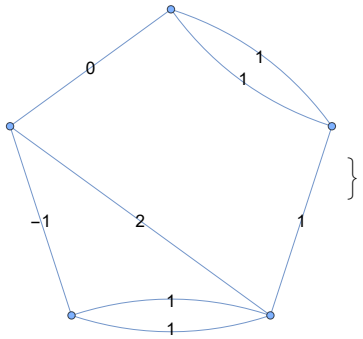
}

{Knot[10, 34],



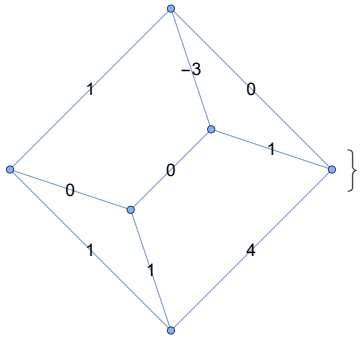
}

{Knot[10, 35],



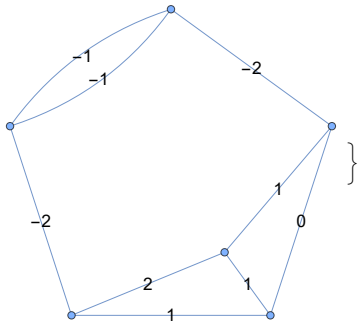
}

{Knot[10, 36],



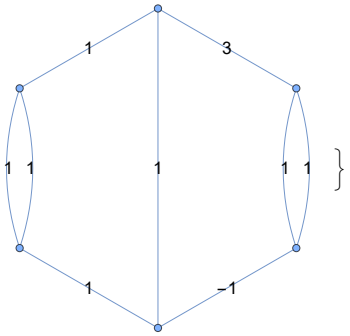
}

{Knot[10, 37],



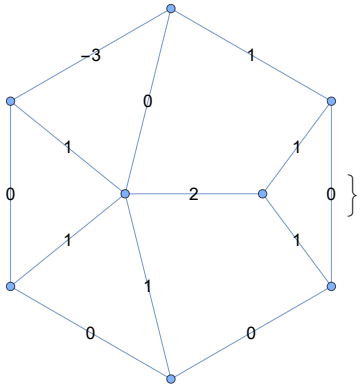
}

{Knot[10, 38],



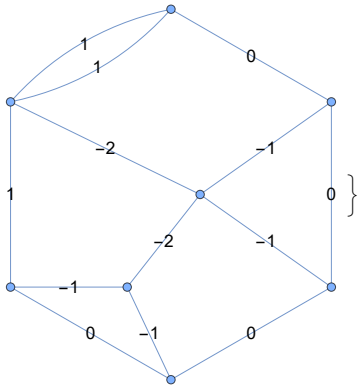
}

{Knot[10, 39],



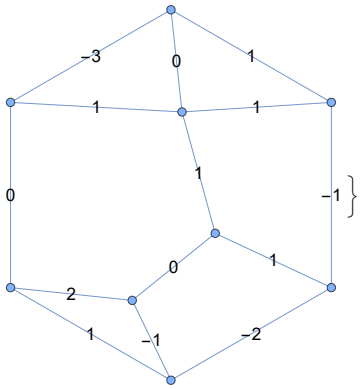
}

{Knot[10, 40],



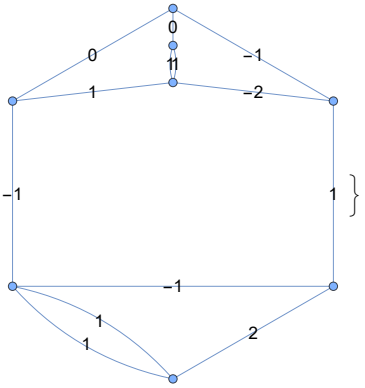
}

{Knot[10, 41],



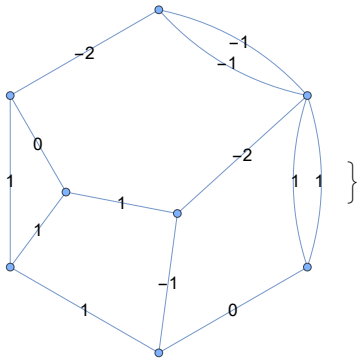
}

{Knot[10, 42],



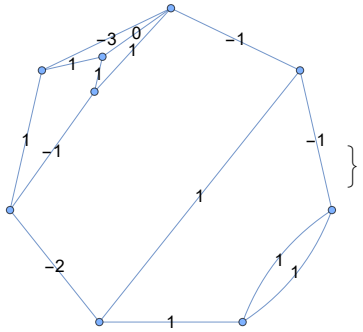
}

{Knot[10, 43],



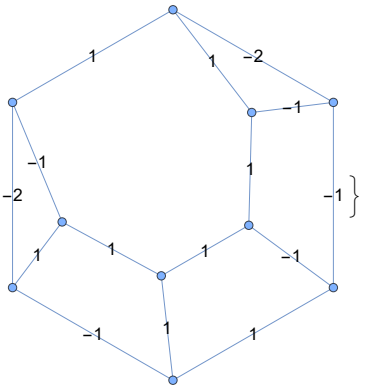
}

{Knot[10, 44],



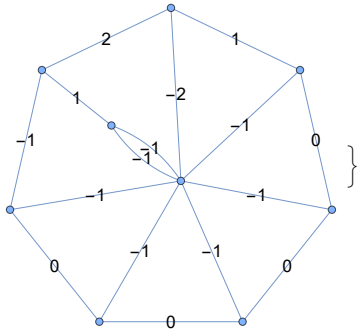
}

{Knot[10, 45],



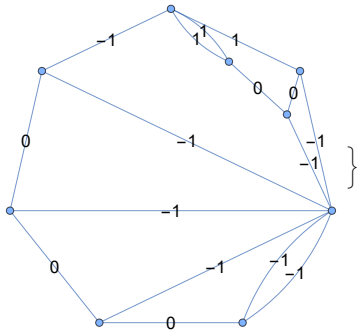
}

{Knot[10, 46],



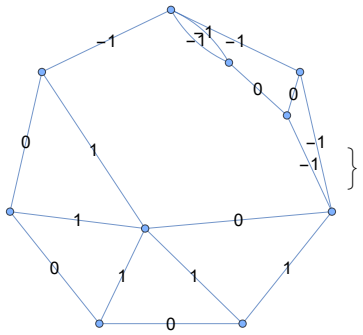
}

{Knot[10, 47],



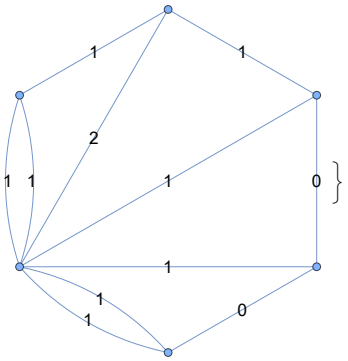
}

{Knot[10, 48],



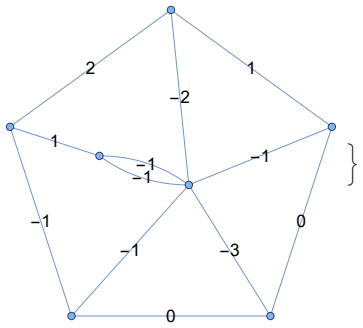
}

{Knot[10, 49],



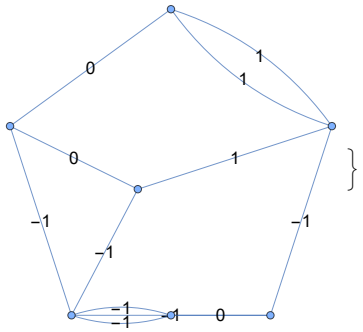
}

{Knot[10, 50],



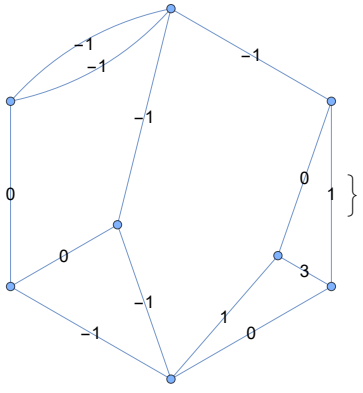
}

{Knot[10, 51],



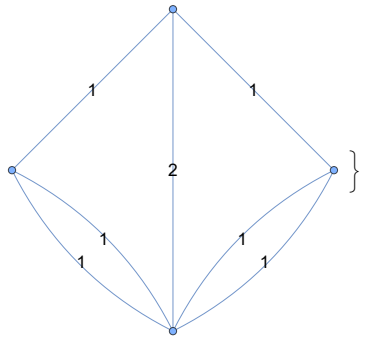
}

{Knot[10, 52],



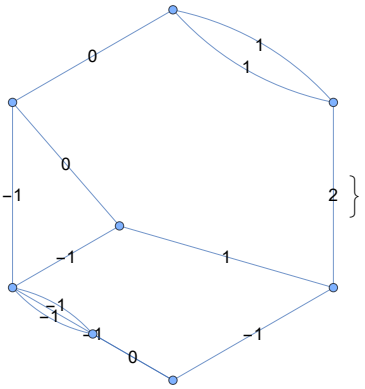
}

{Knot[10, 53],



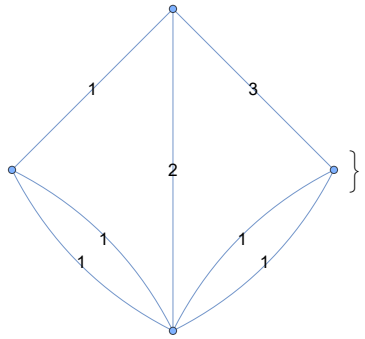
}

{Knot[10, 54],



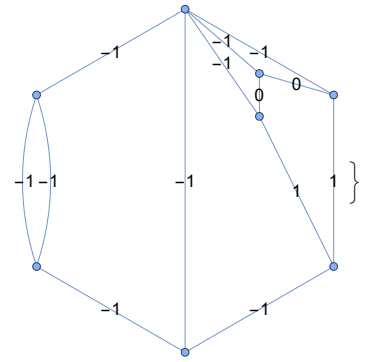
}

{Knot[10, 55],



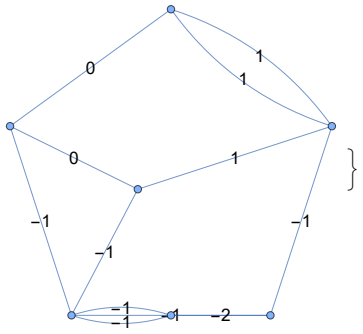
}

{Knot[10, 56],



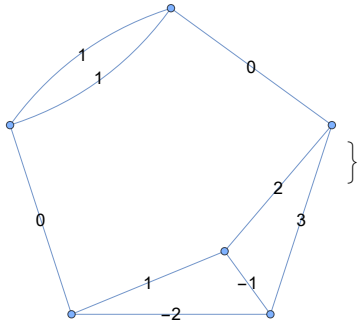
}

{Knot[10, 57],



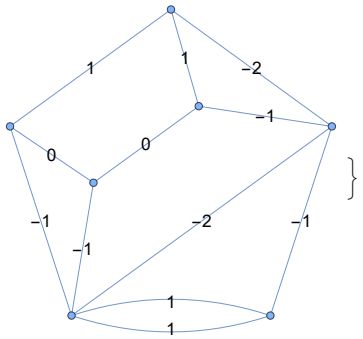
}

{Knot[10, 58],



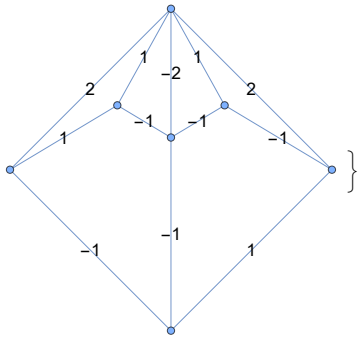
}

{Knot[10, 59],



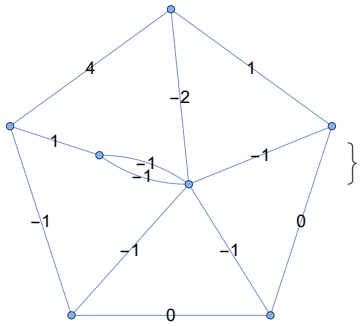
}

{Knot[10, 60],



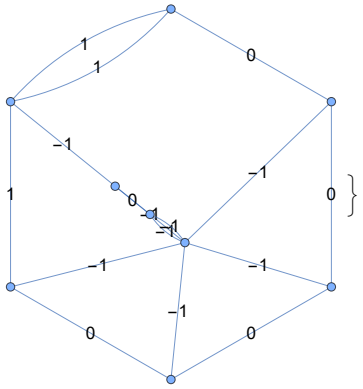
}

{Knot[10, 61],



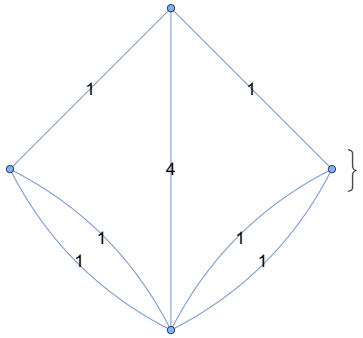
}

{Knot[10, 62],



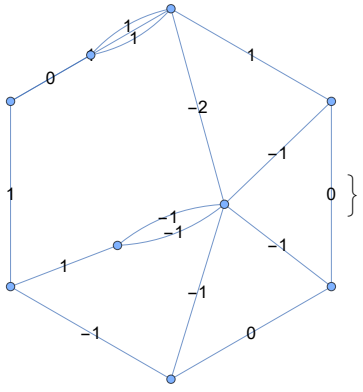
}

{Knot[10, 63],



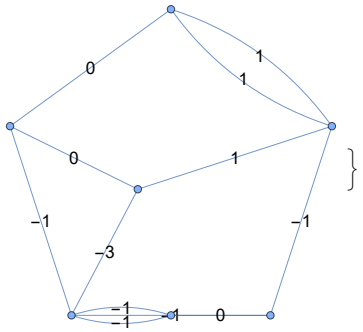
}

{Knot[10, 64],



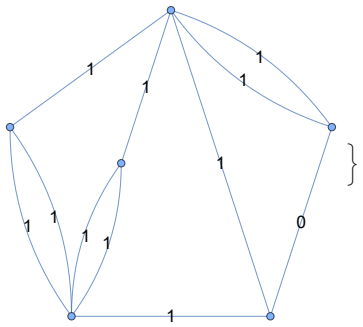
}

{Knot[10, 65],



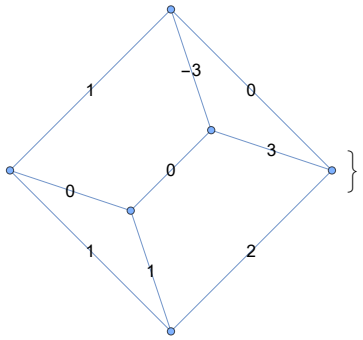
}

{Knot[10, 66],



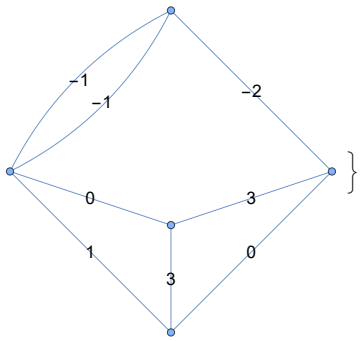
}

{Knot[10, 67],



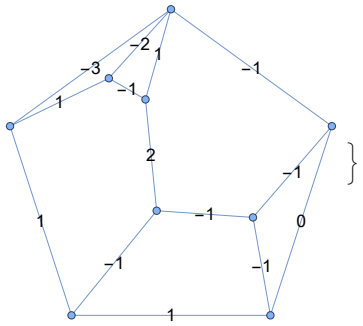
}

{Knot[10, 68],



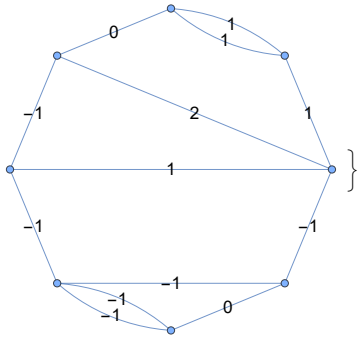
}

{Knot[10, 69],



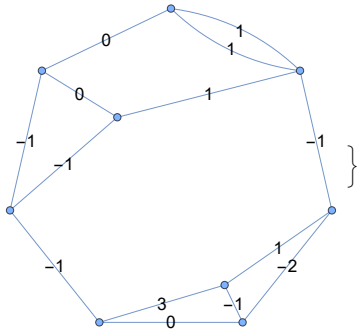
}

{Knot[10, 70],



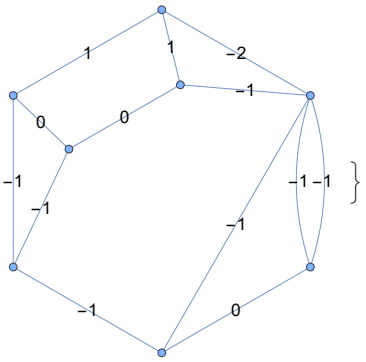
}

{Knot[10, 71],



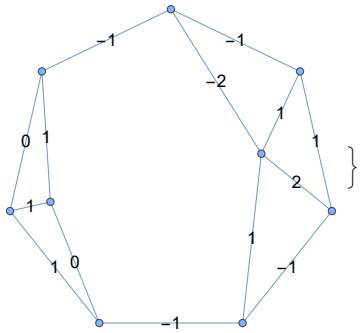
}

{Knot[10, 72],



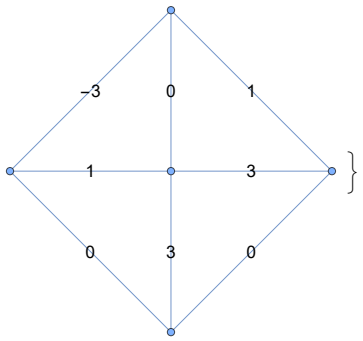
}

{Knot[10, 73],



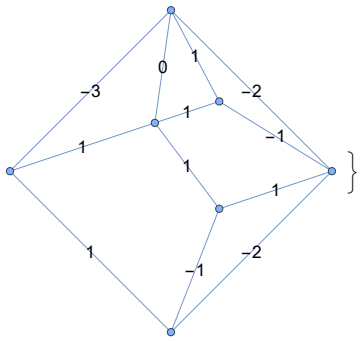
}

{Knot[10, 74],



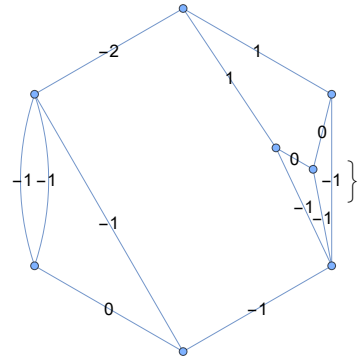
}

{Knot[10, 75],



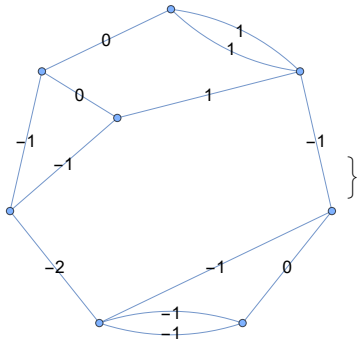
}

{Knot[10, 76],



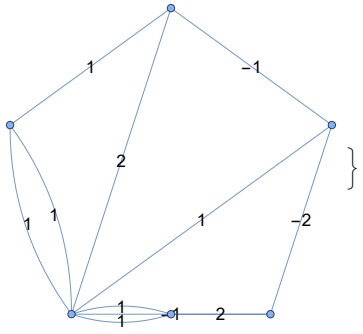
}

{Knot[10, 77],



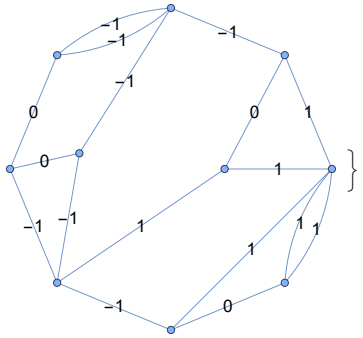
}

{Knot[10, 78],



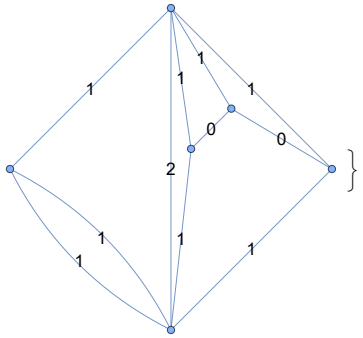
}

{Knot[10, 79],



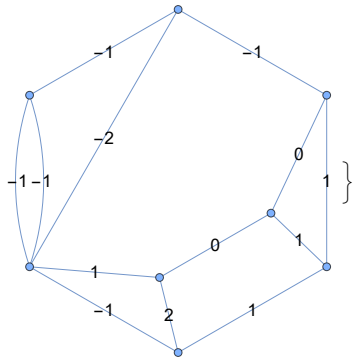
}

{Knot[10, 80],



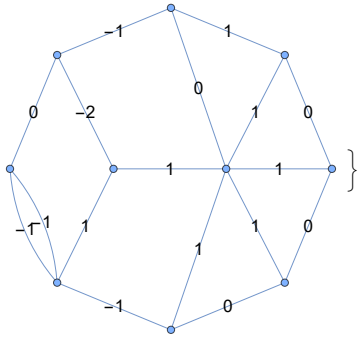
}

{Knot[10, 81],



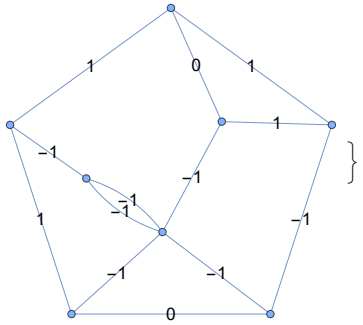
}

{Knot[10, 82],



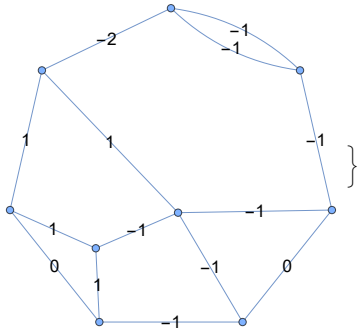
}

{Knot[10, 83],

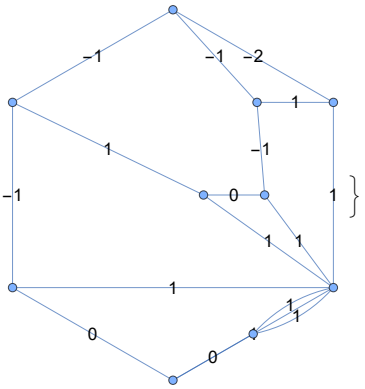


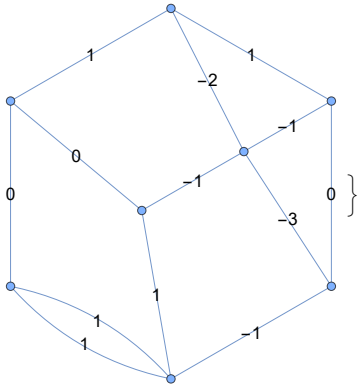
}

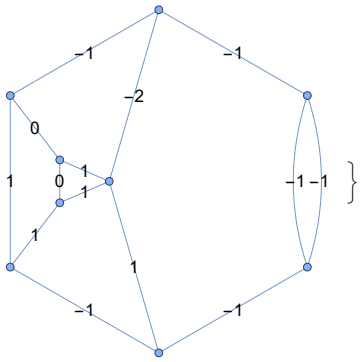
{Knot[10, 84],

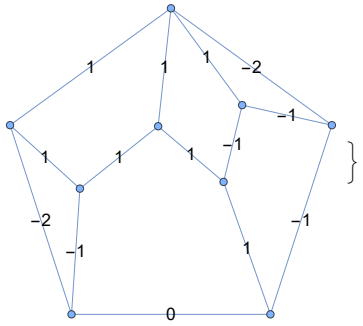


}

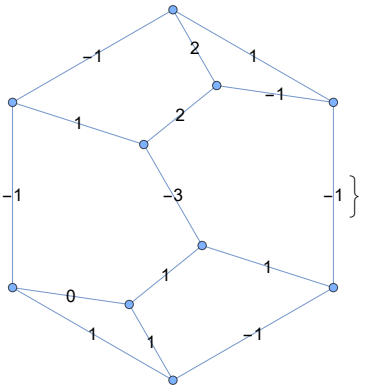
{Knot[10, 85],  }

{Knot[10, 86],  }

{Knot[10, 87],  }

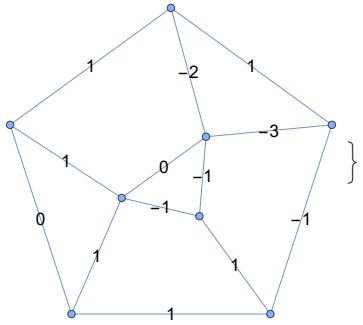
{Knot[10, 88],  }

{Knot[10, 89],



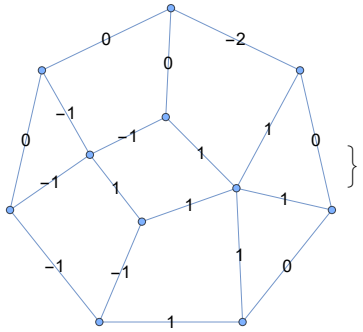
}

{Knot[10, 90],



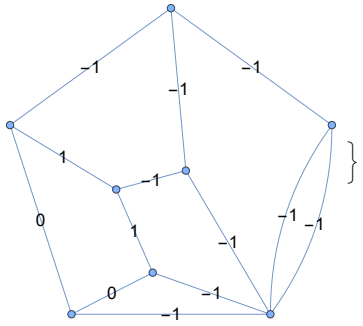
}

{Knot[10, 91],

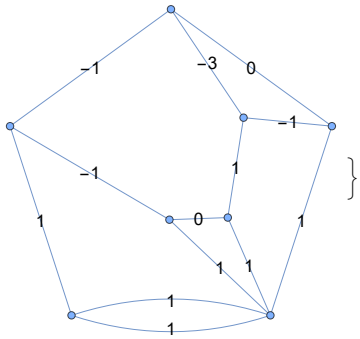


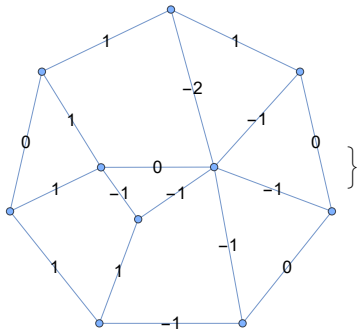
}

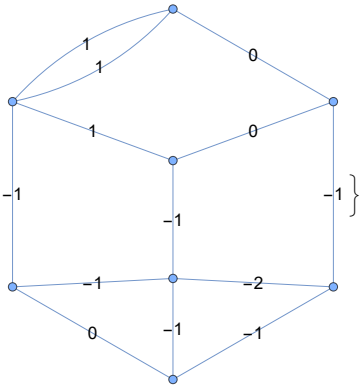
{Knot[10, 92],

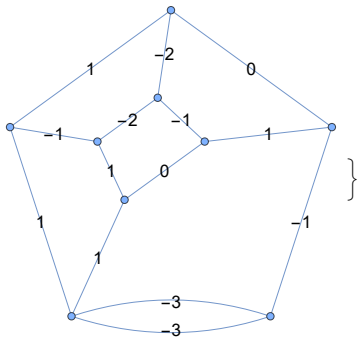


}

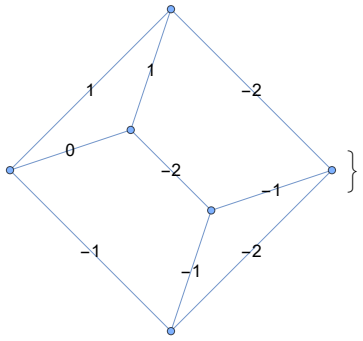
{Knot[10, 93],  }

{Knot[10, 94],  }

{Knot[10, 95],  }

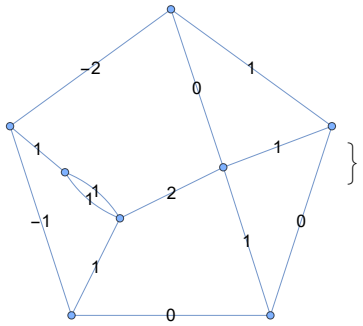
{Knot[10, 96],  }

{Knot[10, 97],



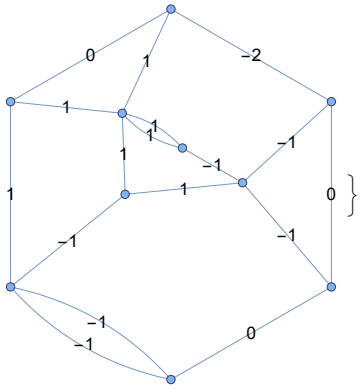
}

{Knot[10, 98],



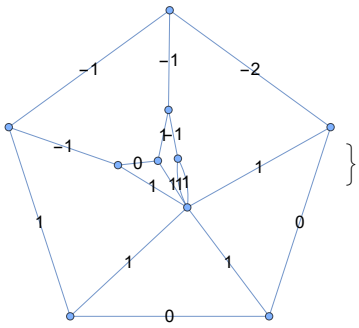
}

{Knot[10, 99],



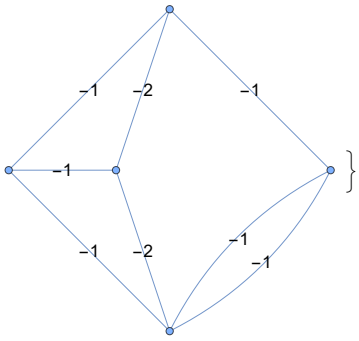
}

{Knot[10, 100],



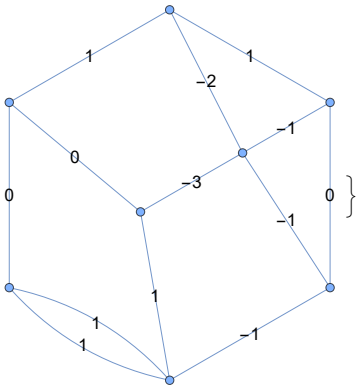
}

{Knot[10, 101],



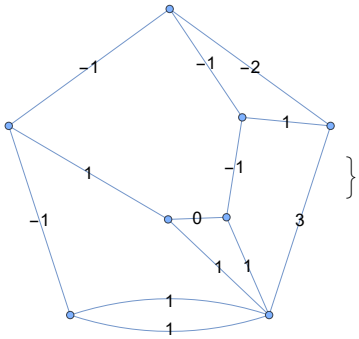
}

{Knot[10, 102],



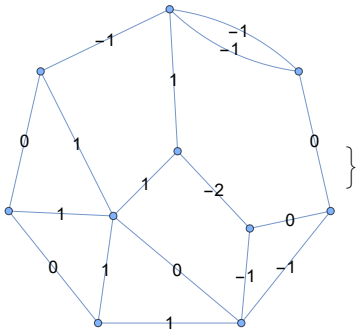
}

{Knot[10, 103],



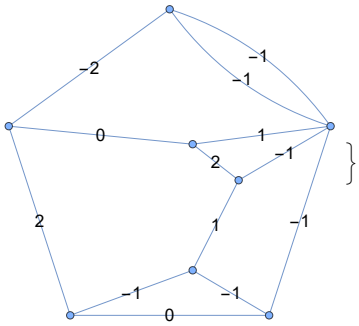
}

{Knot[10, 104],



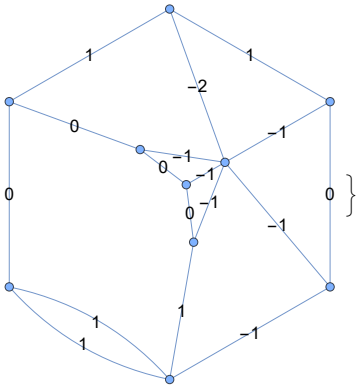
}

{Knot[10, 105],



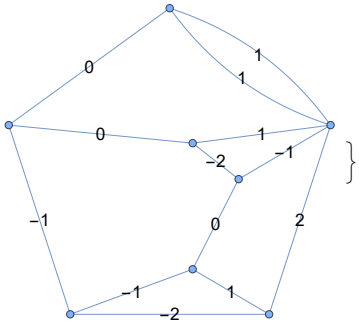
}

{Knot[10, 106],



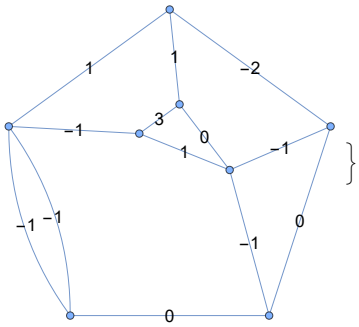
}

{Knot[10, 107],



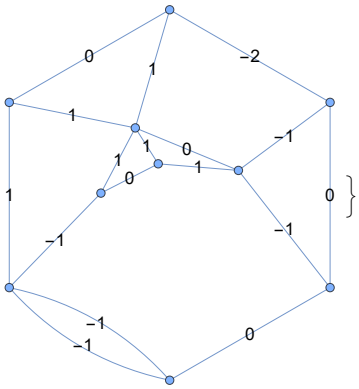
}

{Knot[10, 108],



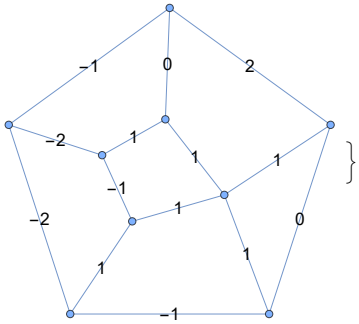
}

{Knot[10, 109],



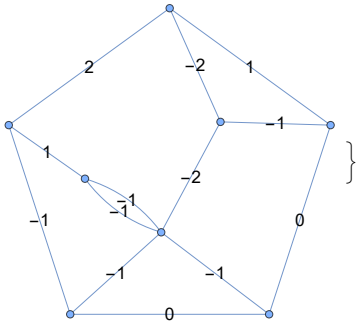
}

{Knot[10, 110],



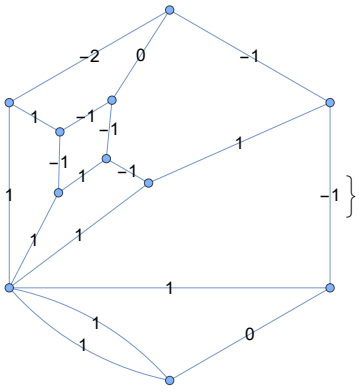
}

{Knot[10, 111],



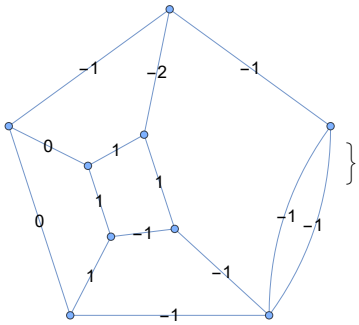
}

{Knot[10, 112],



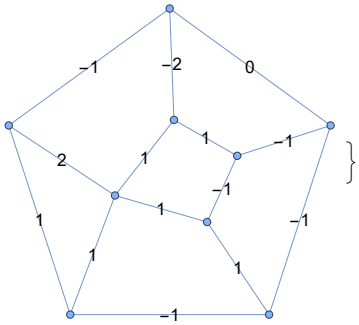
}

{Knot[10, 113],



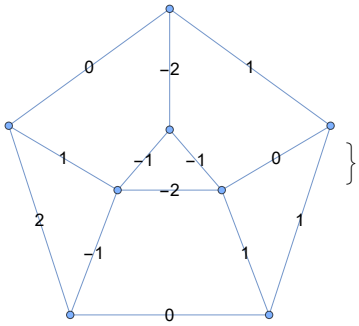
}

{Knot[10, 114],



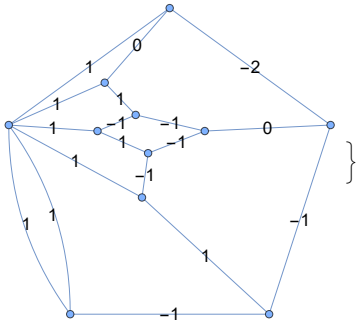
}

{Knot[10, 115],



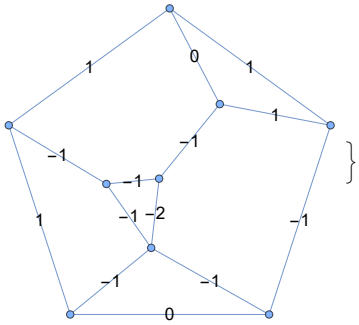
}

{Knot[10, 116],

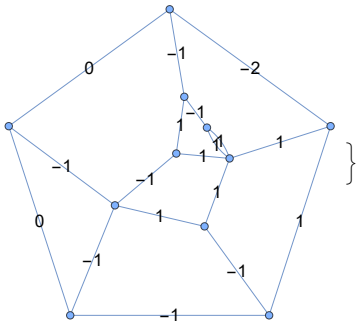


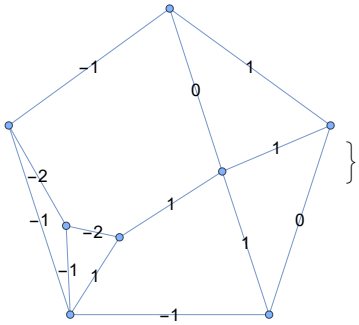
}

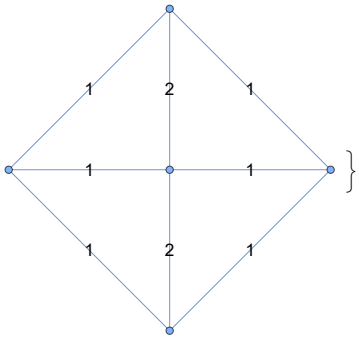
{Knot[10, 117],

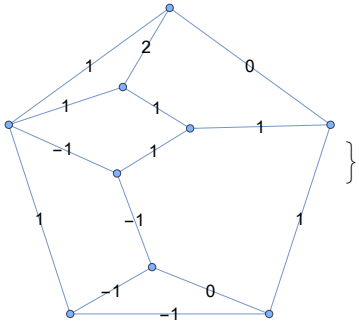


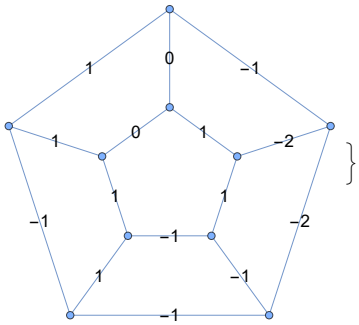
}

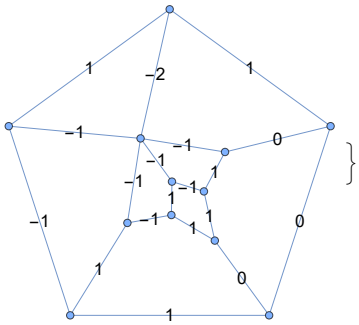
{Knot[10, 118],  }

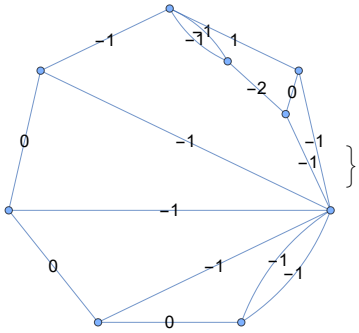
{Knot[10, 119],  }

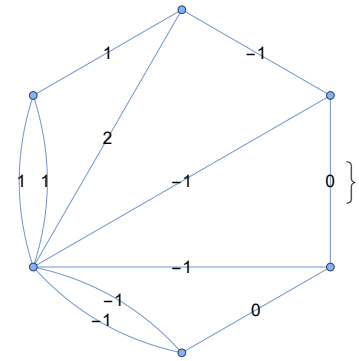
{Knot[10, 120],  }

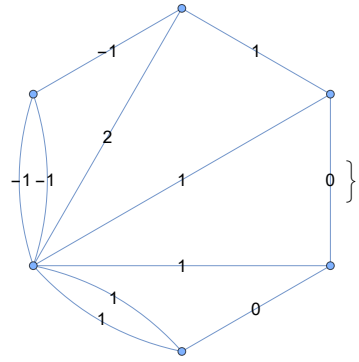
{Knot[10, 121],  }

{Knot[10, 122],  }

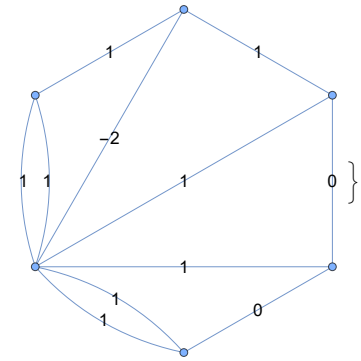
{Knot[10, 123],  }

{Knot[10, 124],  }

{Knot[10, 125],  }

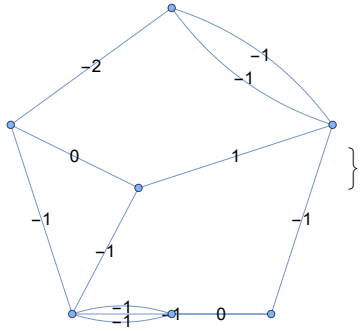
{Knot[10, 126],  }

{Knot[10, 127],



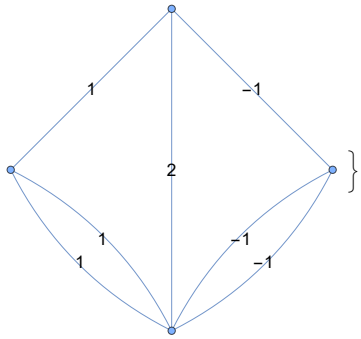
}

{Knot[10, 128],



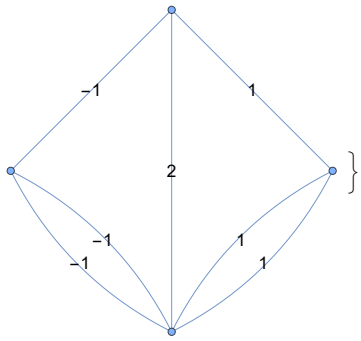
}

{Knot[10, 129],



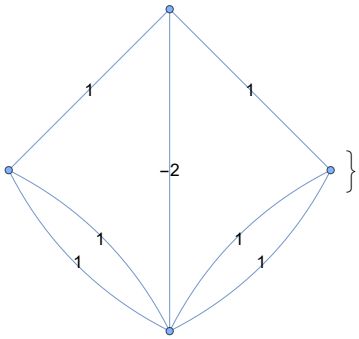
}

{Knot[10, 130],



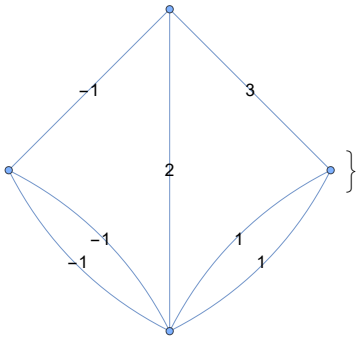
}

{Knot[10, 131],



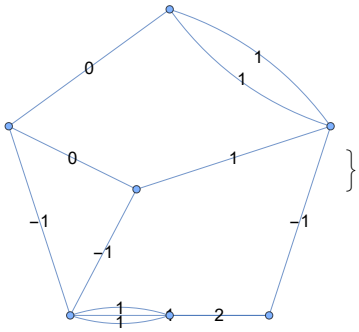
}

{Knot[10, 132],



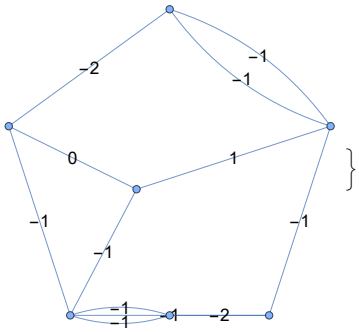
}

{Knot[10, 133],



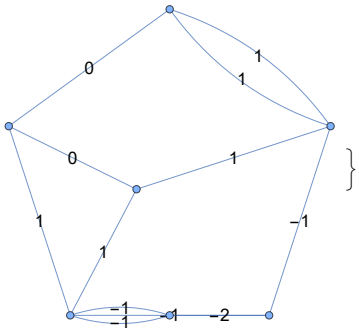
}

{Knot[10, 134],



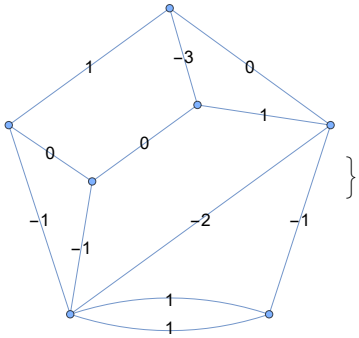
}

{Knot[10, 135],



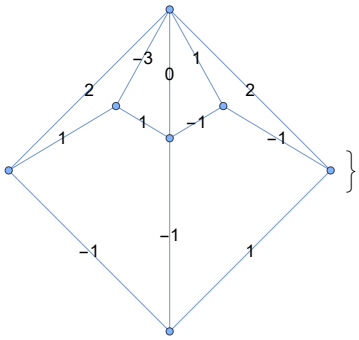
}

{Knot[10, 136],



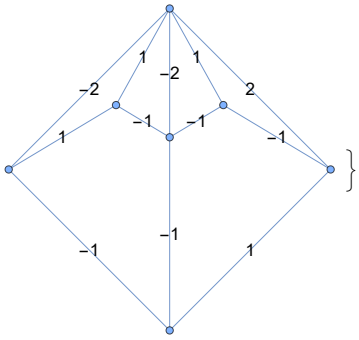
}

{Knot[10, 137],



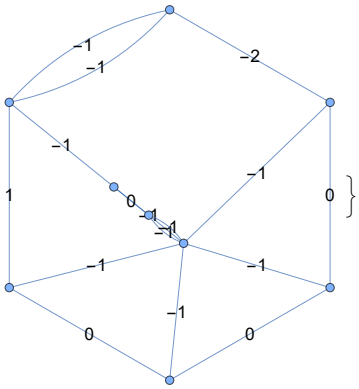
}

{Knot[10, 138],



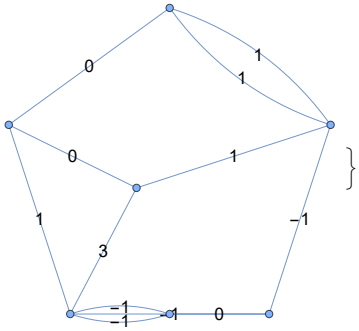
}

{Knot[10, 139],



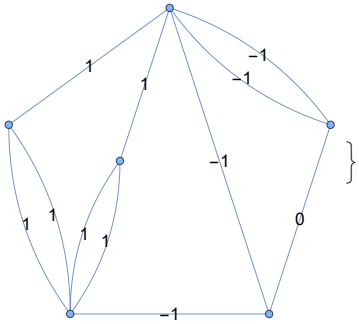
}

{Knot[10, 140],



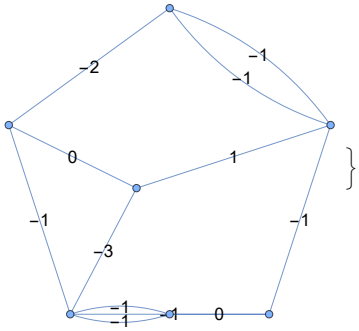
}

{Knot[10, 141],



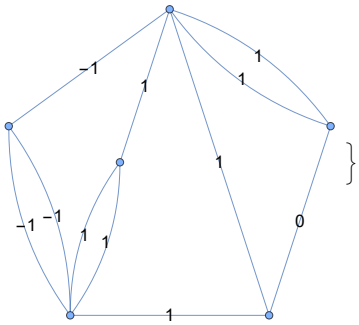
}

{Knot[10, 142],



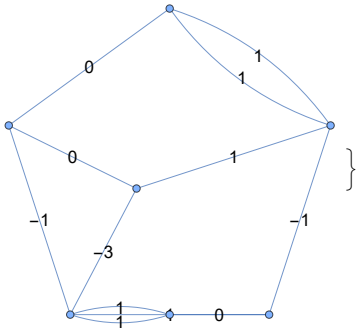
}

{Knot[10, 143],



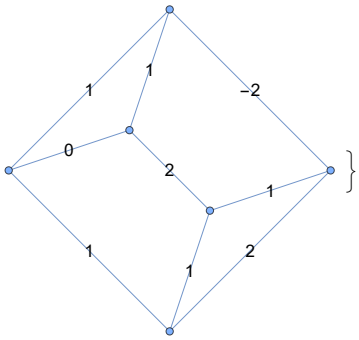
}

{Knot[10, 144],



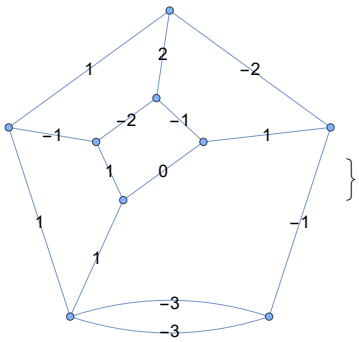
}

{Knot[10, 145],



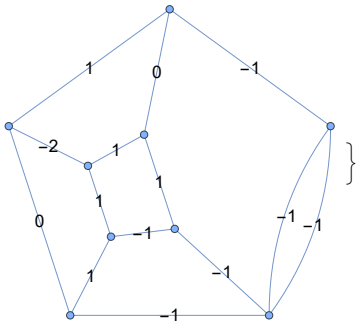
}

{Knot[10, 146],



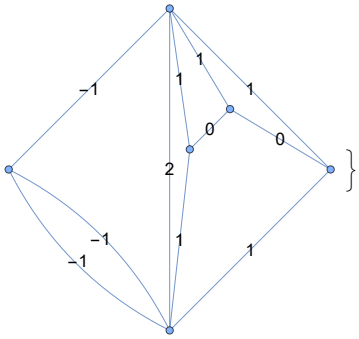
}

{Knot[10, 147],



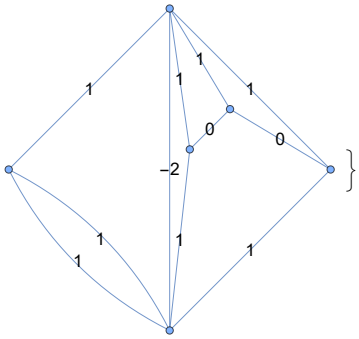
}

{Knot[10, 148],



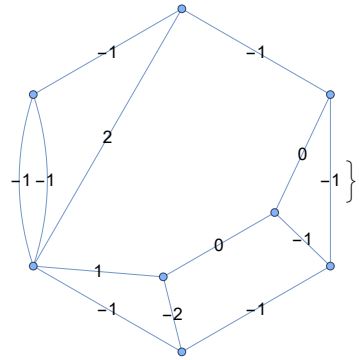
}

{Knot[10, 149],

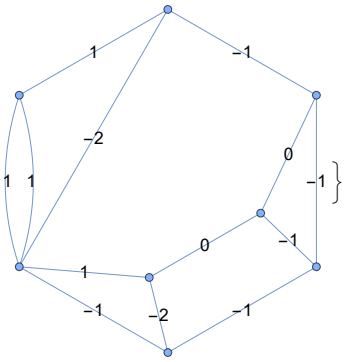


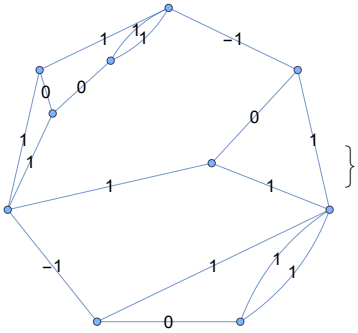
}

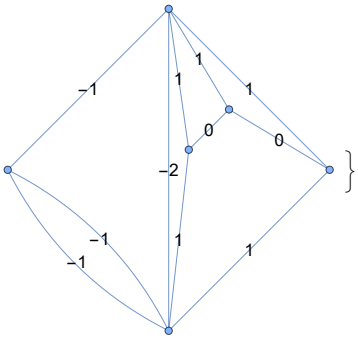
{Knot[10, 150],

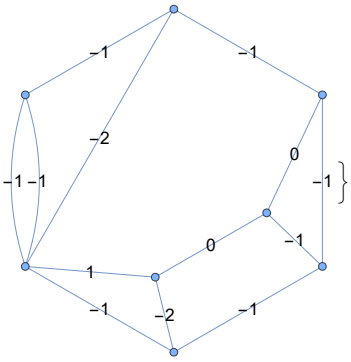


}

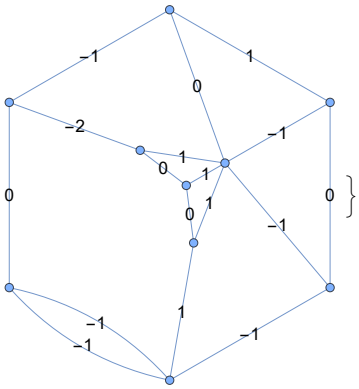
{Knot[10, 151],  }

{Knot[10, 152],  }

{Knot[10, 153],  }

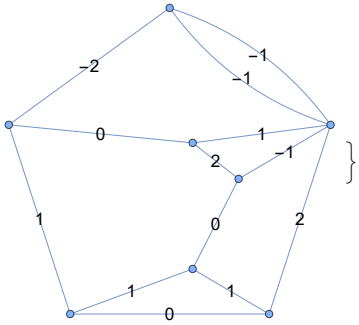
{Knot[10, 154],  }

{Knot[10, 155],



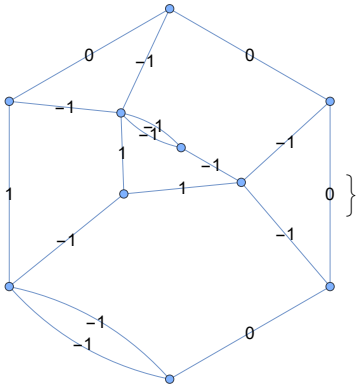
}

{Knot[10, 156],



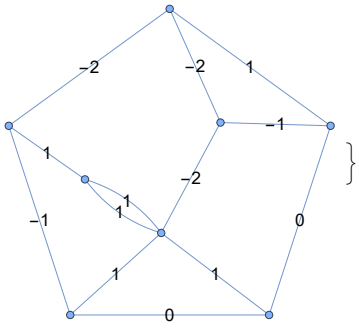
}

{Knot[10, 157],



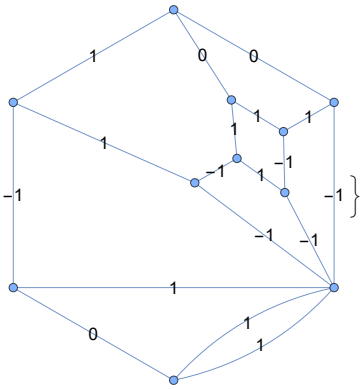
}

{Knot[10, 158],



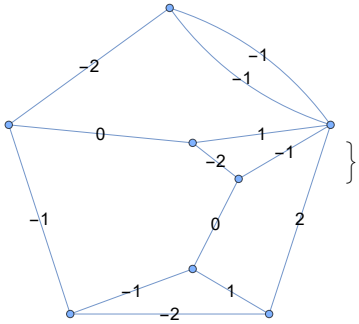
}

{Knot[10, 159],



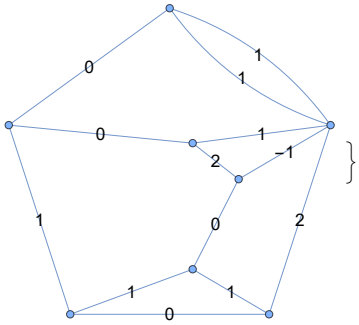
}

{Knot[10, 160],



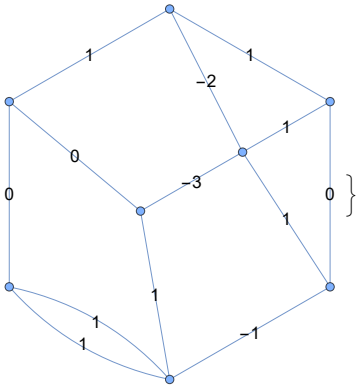
}

{Knot[10, 161],



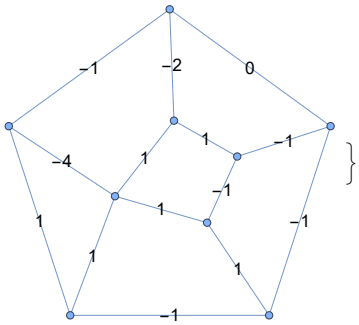
}

{Knot[10, 162],



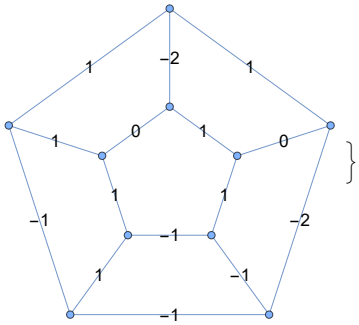
}

{Knot[10, 163],



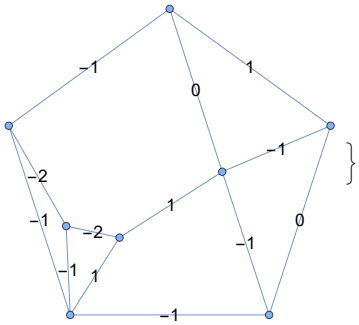
}

{Knot[10, 164],



}

{Knot[10, 165],



}