(Alt) In[ ]:=

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\Projects\\HigherRank\\DunfieldKnots"];
Once[<< KnotTheory`];
<< ../Rot.m
T3 = T1 T2;
```

```
C:\drorbn\AcademicPensieve\Projects\KnotTheory\KnotTheory

Loading KnotTheory` version of February 2, 2020, 10:53:45.2097.
Read more at http://katlas.org/wiki/KnotTheory.
```

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

```
Loading Rot.m from http://drorbn.net/AP/Projects/HigherRank to compute rotation numbers.
```

(Alt) In[ ]:=

```
CCF[ℰ_] := ExpandDenominator@ExpandNumerator@Together[ℰ];
CCF[ℰ_] := Factor[ℰ];
CF[ℰ_List] := CF /@ ℰ;
CF[ℰ_] := Module[{vs = Cases[ℰ, (x | p | π | g)__, ∞] ⋃ {x, p, ϵ}, ps, c},
   Total[CoefficientRules[Expand[ℰ], vs] /. (ps_ → c_) :> CCF[c] (Times @@ vs^ps)]];
```

(Alt) In[ ]:=

$$R_1[1, i\_, j\_] = CF[\; 1/2 - T_3\, g_{1ji}\, g_{2ji} - g_{3ii} + g_{2jj}\, g_{3ii} + T_1\,(T_3 - 1)\, g_{1ji}\, g_{3ji} + T_2\,(T_3 - 1)\, g_{2ji}\, g_{3ji} - T_2\, g_{2ji}\, g_{3jj} + (g_{1jj}\, g_{2ii} + (T_3 - 1)\, g_{1jj}\, g_{2ji} - T_1\, g_{1ii}\, g_{2jj} - g_{1jj}\, g_{3ii} - T_1\,(T_3 - 1)\, g_{1jj}\, g_{3ji} + T_1\, g_{1ii}\, g_{3jj}) / (T_1 - 1)];$$

(Alt) In[ ]:=

$$R_1[-1, i\_, j\_] = CF\Big[-1/2 - T_1^{-1} g_{1ji}\, g_{2ii} - \left(1 - T_1^{-1} - T_2^{-1}\right) g_{1ji}\, g_{2ji} - g_{1jj}\, g_{2ji} - g_{1ji}\, g_{2jj} + g_{3ii} + T_1^{-1} g_{1ji}\, g_{3ii} - \left(1 - T_2^{-1}\right) g_{2ji}\, g_{3ii} - g_{2jj}\, g_{3ii} + \left(1 - T_3^{-1}\right) g_{1ji}\, g_{3ji} - \left(1 - T_3^{-1}\right) g_{2ii}\, g_{3ji} + \left(2 - T_2^{-1}\right)\left(1 - T_3^{-1}\right) g_{2ji}\, g_{3ji} + \left(1 - T_3^{-1}\right) g_{2jj}\, g_{3ji} + g_{1ji}\, g_{3jj} + g_{2ji}\, g_{3jj} + \Big(T_1 \left(1 - T_2^{-1}\right) g_{1ii}\, g_{2ji} - g_{1jj}\, g_{2ii} + T_1\, g_{1ii}\, g_{2jj} + g_{1jj}\, g_{3ii} - T_2^{-1} (T_3 - 1)\, g_{1ii}\, g_{3ji} - T_1\, g_{1ii}\, g_{3jj}\Big) \Big/ (T_1 - 1)\Big];$$

(Alt) In[ ]:=

$$\theta[\{1, i0\_, j0\_\}, \{1, i1\_, j1\_\}] = -T_1\,(T_3 - 1)\, g_{1,j1,i0}\, g_{2,i1,i0}\, g_{3,j0,i1} + (T_3 - 1)\, g_{1,j1,j0}\, g_{2,i1,i0}\, g_{3,j0,i1} + T_1\,(T_3 - 1)\, g_{1,j1,i0}\, g_{2,j1,i0}\, g_{3,j0,i1} - (T_3 - 1)\, g_{1,j1,j0}\, g_{2,j1,i0}\, g_{3,j0,i1};$$

(Alt) In[ ]:=

$$\theta[\{1, i0\_, j0\_\}, \{-1, i1\_, j1\_\}] = (T_3 - 1)\, g_{1,j1,i0}\, g_{2,i1,i0}\, g_{3,j0,i1} - T_1^{-1}(T_3 - 1)\, g_{1,j1,j0}\, g_{2,i1,i0}\, g_{3,j0,i1} - (T_3 - 1)\, g_{1,j1,i0}\, g_{2,j1,i0}\, g_{3,j0,i1} + T_1^{-1}(T_3 - 1)\, g_{1,j1,j0}\, g_{2,j1,i0}\, g_{3,j0,i1};$$

(Alt) In[ ]:=

```
θ[{-1, i0_, j0_}, {1, i1_, j1_}] = CF[
   T1^-1 T2^-1 (T3 - 1) (g1,j1,i0 g2,i1,i0 g3,j0,i1 -
      T1 g1,j1,j0 g2,i1,i0 g3,j0,i1 - g1,j1,i0 g2,j1,i0 g3,j0,i1 + T1 g1,j1,j0 g2,j1,i0 g3,j0,i1)];
```

(Alt) In[ ]:=

```
θ[{-1, i0_, j0_}, {-1, i1_, j1_}] = CF[
   (1 - T3^-1) (-T1^-1 g1,j1,i0 g2,i1,i0 g3,j0,i1 +
      g1,j1,j0 g2,i1,i0 g3,j0,i1 + T1^-1 g1,j1,i0 g2,j1,i0 g3,j0,i1 - g1,j1,j0 g2,j1,i0 g3,j0,i1)];
```

(Alt) In[ ]:=

```
Γ1[φ_, k_] = -φ / 2 + φ g3,k,k;
```

(Alt) In[ ]:=

```
θ[K_] := Module[{Cs, φ, n, A, s, i, j, k, Δ, G, ν, α, β, gEval, c, z},
   {Cs, φ} = Rot[K]; n = Length[Cs];
   A = IdentityMatrix[2 n + 1];
   Cases[Cs, {s_, i_, j_} :> (A[[{i, j}, {i + 1, j + 1}]] += ( -T^s  T^s - 1
                                                                 0     -1    ))];
   Δ = T^((-Total[φ] - Total[Cs[[All,1]]]) / 2) Det[A];
   G = Inverse[A]; gEval[ℰ_] := Factor[ℰ /. g_{ν_,α_,β_} :> (G[[α, β]] /. T → T_ν)];
   z = gEval[Sum_{k1=1}^{n} Sum_{k2=1}^{n} θ[Cs[[k1]], Cs[[k2]]]];
   z += gEval[Sum_{k=1}^{n} R1 @@ Cs[[k]]];
   z += gEval[Sum_{k=1}^{2 n} Γ1[φ[[k]], k]];
   {Δ, (Δ /. T → T1) (Δ /. T → T2) (Δ /. T → T3) z} // Factor ];
```

(Alt) In[ ]:=

```
PolyPlot[0] = Graphics[{}];
PolyPlot[p_] := Module[{crs, m1, m2, maxc, minc, s, hex},
  crs = CoefficientRules[T1^(m1=-Exponent[p,T1,Min]) T2^(m2=-Exponent[p,T2,Min]) p, {T1, T2}];
  maxc = N@Log@Max@Abs[Last /@ crs];
  minc = N@Log@Min@Select[Abs[Last /@ crs], # > 0 &];
  If[minc == maxc, s[_] = 0, s[c_] := s[c] = (maxc - Log@c) / (maxc - minc)];
  hex = Table[{Cos[α], Sin[α]} / Cos[2 π / 12] / 2, {α, 2 π / 12, 2 π, 2 π / 6}];
  Graphics[crs /. ({x1_, x2_} → c_) :→ {
     If[c == 0, White, Lighter[Which[
        c > 0 ∧ OddQ[c], Orange,
        c > 0 ∧ EvenQ[c], Red,
        c < 0 ∧ OddQ[c], Green,
        c < 0 ∧ EvenQ[c], Blue
       ], 0.88 s[Abs@c]]],
     Polygon[(({{1, -1/2}, {0, √3/2}}).{x1 + m1, x2 + m2} + #) & /@ hex] }]];
PolyPlot[{Δ_, θ_}] := PolyPlot[θ]
```

In[ ]:=

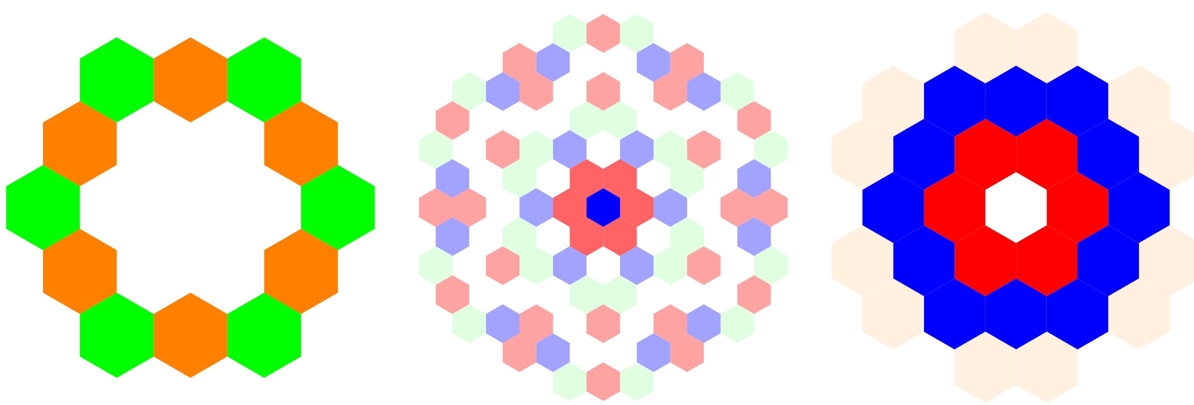
```
GraphicsRow[PolyPlot[θ[Knot[#]]][[2]] &
  /@ {"3_1", "K11n34", "K11n42"}]
```

KnotTheory: Loading precomputed data in PD4Knots`.

KnotTheory: Loading precomputed data in DTCode4KnotsTo11`.

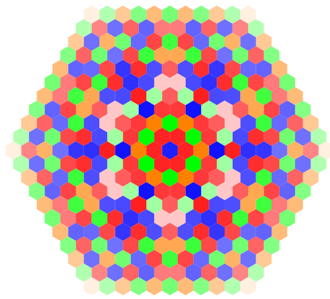
KnotTheory: The GaussCode to PD conversion was written by Siddarth Sankaran at the University of Toronto in the summer of 2005.

Out[ ]=



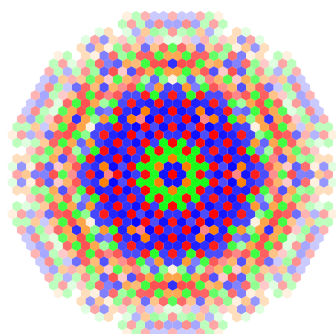
In[∘]:= **AbsoluteTiming@PolyPlot@θ@Knot[15, Alternating, 1000]**

Out[∘]=

{0.408887,  }

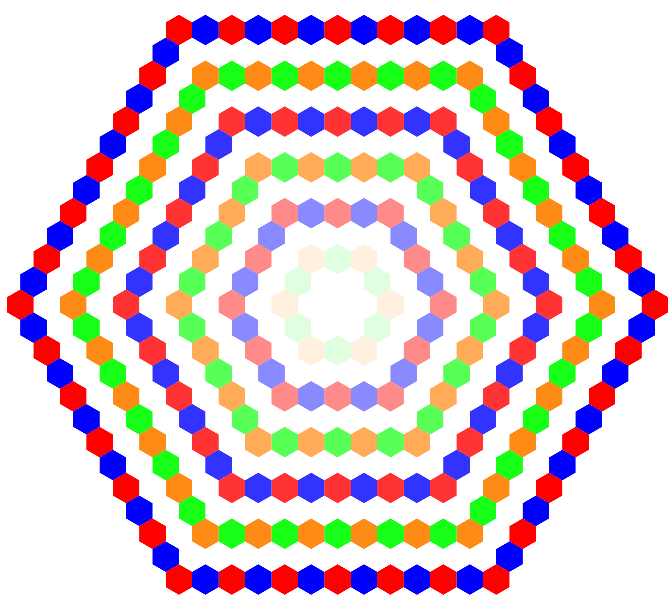
In[∘]:= **AbsoluteTiming@**
PolyPlot[θ[EPD[$X_{14,1}$, $\bar{X}_{2,29}$, $X_{3,40}$, $X_{43,4}$, $\bar{X}_{26,5}$, $X_{6,95}$, $X_{96,7}$, $X_{13,8}$, $\bar{X}_{9,28}$, $X_{10,41}$, $X_{42,11}$, $\bar{X}_{27,12}$,
$X_{30,15}$, $\bar{X}_{16,61}$, $\bar{X}_{17,72}$, $\bar{X}_{18,83}$, $X_{19,34}$, $\bar{X}_{89,20}$, $\bar{X}_{21,92}$, $\bar{X}_{79,22}$, $\bar{X}_{68,23}$, $\bar{X}_{57,24}$, $\bar{X}_{25,56}$, $X_{62,31}$,
$X_{73,32}$, $X_{84,33}$, $\bar{X}_{50,35}$, $X_{36,81}$, $X_{37,70}$, $X_{38,59}$, $\bar{X}_{39,54}$, $X_{44,55}$, $X_{58,45}$, $X_{69,46}$, $X_{80,47}$, $X_{48,91}$,
$X_{90,49}$, $X_{51,82}$, $X_{52,71}$, $X_{53,60}$, $\bar{X}_{63,74}$, $\bar{X}_{64,85}$, $\bar{X}_{76,65}$, $\bar{X}_{87,66}$, $\bar{X}_{67,94}$, $\bar{X}_{75,86}$, $\bar{X}_{88,77}$, $\bar{X}_{78,93}$**]]⟦2⟧]**

Out[∘]=

{11.9446,  }

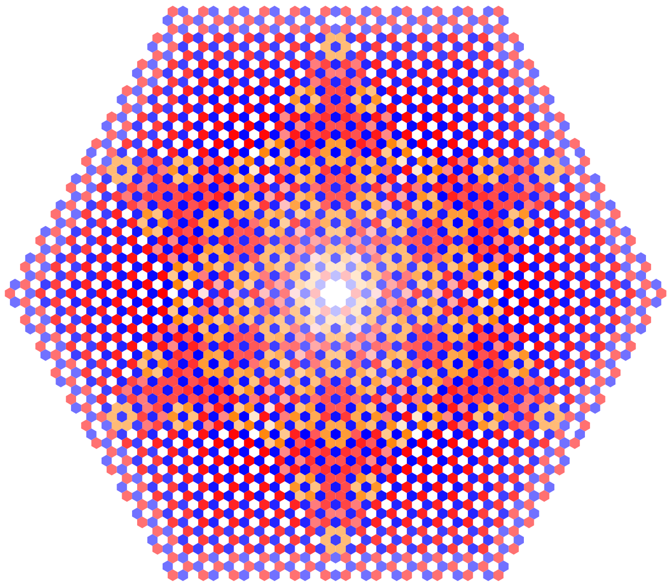
In[∘]:= **PolyPlot[θ[TorusKnot[13, 2]]]**

Out[∘]=



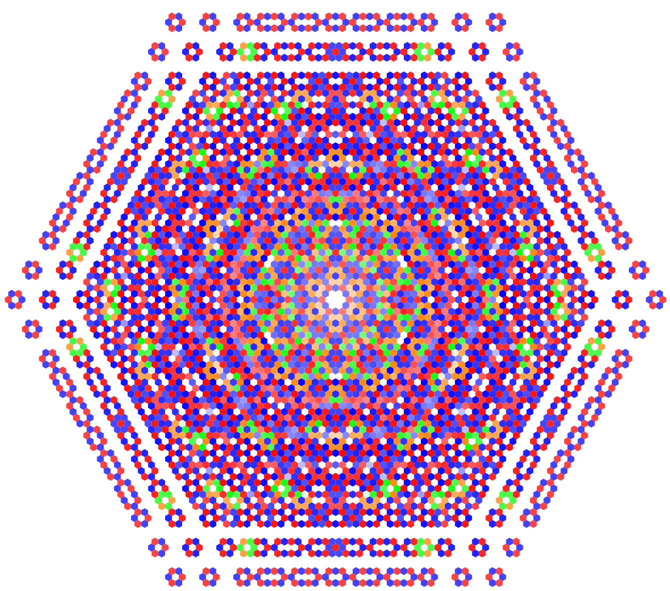
In[∘]:= PolyPlot[θ[TorusKnot[17, 3]]]

Out[∘]=



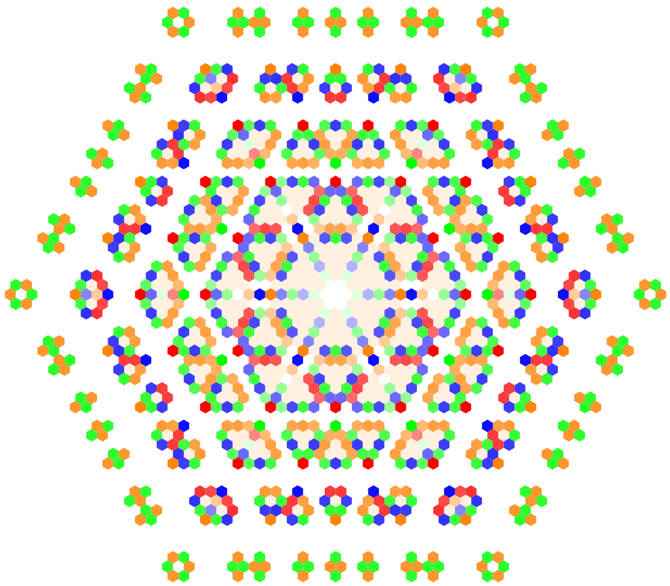
In[∘]:= PolyPlot[θ[TorusKnot[13, 5]]]

Out[∘]=



In[•]:= PolyPlot[θ[TorusKnot[7, 6]]]

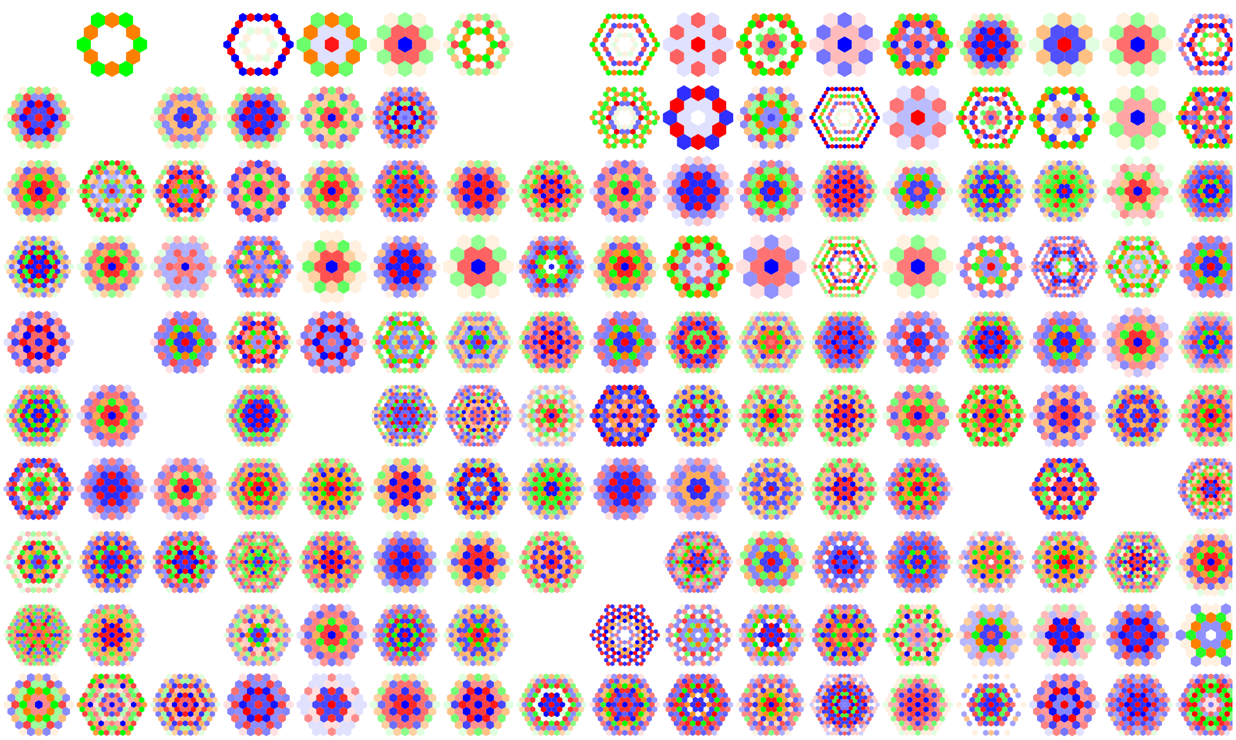
Out[•]=



In[•]:= tab250 = {0}~Join~Table[θ[K]⟦2⟧, {K, AllKnots[{3, 10}]}];

In[•]:= g250 = GraphicsGrid[Partition[PolyPlot /@ tab250, 25], Spacings → 0]

Out[•]=



(Alt) In[ ]:=

```
DunfieldKnots =
  ReadList["../../../People/Dunfield/nmd_random_knots"] /. k_Integer :→ k + 1;
DK[n_] := DunfieldKnots[[n - 2]];
```

(Alt) In[ ]:=

```
DKString[n_] := StringDrop[ToString[1000 + n], 1]
```

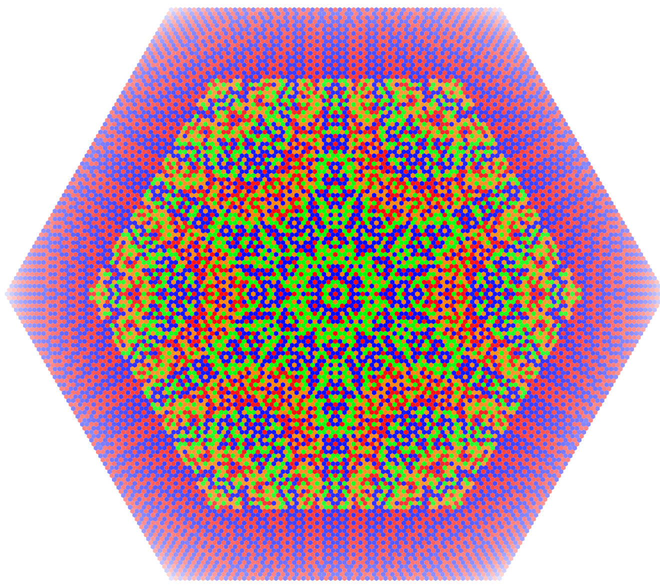
(Alt) In[ ]:=

```
PP[n_] := If[FileExistsQ[from = "D" <> DKString[n] <> ".m"],
  PolyPlot[Get[from][[2]] /. {T1 → T₁, T2 → T₂}]
 ]
```

In[ ]:=

```
PP[150]
```

Out[ ]=



(Alt) In[ ]:=

```
Monitor[
 Do[
  If[FileExistsQ["D" <> DKString[n] <> ".m"] ∧
    Not@FileExistsQ["PC" <> DKString[n] <> ".png"],
   {at, th} = Get["D" <> DKString[n] <> ".m"];
   pc = PolyPlot[th /. {T1 → T₁, T2 → T₂}];
   Export["PC" <> DKString[n] <> ".png", pc];
  ],
  {n, 3, 1000, 1}
 ],
 {n, at, pc}
]
```

(Alt) In[ ]:=

```
PolyPlot[0] = Graphics[{}];
PolyPlot[p_] := Module[{crs, m1, m2, maxc, minc, s, hex},
  crs = CoefficientRules[T1^(m1=-Exponent[p,T1,Min]) T2^(m2=-Exponent[p,T2,Min]) p, {T1, T2}];
  maxc = N@Log@Max@Abs[Last /@ crs];
  minc = N@Log@Min@Select[Abs[Last /@ crs], # > 0 &];
  If[minc == maxc, s[_] = 0, s[c_] := s[c] = (maxc - Log@c) / (maxc - minc)];
  hex = Table[{Cos[α], Sin[α]} / Cos[2 π / 12] / 2, {α, 2 π / 12, 2 π, 2 π / 6}];
  Graphics[crs /. ({x1_, x2_} → c_) :→ {
     If[c == 0, White, Lighter[Which[
        c > 0 ∧ OddQ[c], Orange,
        c > 0 ∧ EvenQ[c], Green,
        c < 0 ∧ OddQ[c], Orange,
        c < 0 ∧ EvenQ[c], Green
       ], 0.88 s[Abs@c]]],
     Polygon[(({{1, -1/2}, {0, √3/2}}).{x1 + m1, x2 + m2} + #) & /@ hex]}]];
PolyPlot[{Δ_, θ_}] := PolyPlot[θ]
```

(Alt) In[ ]:=

```
PP[150]
```

(Alt) Out[ ]=