Code for the 32 crystallographic Groups

This is given in GS Chapter 6 but to make this list self contained here is the code . This should be initialized before running these notebooks in **Mathematica.**

In[15]:=
```
DBbox[α_] := {GrayLevel[.3], EdgeForm[],
  Polyhedron[α /@ {{{-1, -.5, .5}, {-1, .5, .5}, {-1, .5, -.5}, {-1, -.5, -.5}},
     {{-1, -.5, .5}, {-1, -.5, -.5}, {1, -.5, -.5}, {1, -.5, .5}},
     {{-1, -.5, -.5}, {1, -.5, -.5}, {1, .5, -.5}, {-1, .5, -.5}},
     {{-1, .5, .5}, {1, .5, .5}, {1, .5, -.5}, {-1, .5, -.5}},
     {{1, .5, .5}, {1, .5, -.5}, {1, -.5, -.5}, {1, -.5, .5}},
     {{1, .5, .5}, {1, -.5, .5}, {-1, -.5, .5}, {-1, .5, .5}}}], {Blue, EdgeForm[],
   Polygon[α@{{-.8, 0, .52}, {0, .3, .52}, {.8, 0, .52}, {0, -.3, .52}}]},
  (*New polygons*)
  {Orange, EdgeForm[], Polygon[α@{{-.8, 0, .52}, {-1, -.5, .52}, {-1, .5, .52}}],
   Polygon[α@{{.8, 0, .52}, {1, -.5, .52}, {1, .5, .52}}],
   Polygon[α@{{-1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@{{1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@{{-1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@{{1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],

   Polygon[α@*σ4x@{{-.8, 0, .52}, {-1, -.5, .52}, {-1, .5, .52}}],
   Polygon[α@*σ4x@{{.8, 0, .52}, {1, -.5, .52}, {1, .5, .52}}],
   Polygon[α@*σ4x@{{-1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*σ4x@{{1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*σ4x@{{-1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@*σ4x@{{1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@*ηx@{{-.8, 0, .52}, {-1, -.5, .52}, {-1, .5, .52}}],
   Polygon[α@*ηx@{{.8, 0, .52}, {1, -.5, .52}, {1, .5, .52}}],
   Polygon[α@*ηx@{{-1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*ηx@{{1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*ηx@{{-1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@*ηx@{{1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@*ηx@*σ4x@{{-.8, 0, .52}, {-1, -.5, .52}, {-1, .5, .52}}],
   Polygon[α@*ηx@σ4x@{{.8, 0, .52}, {1, -.5, .52}, {1, .5, .52}}],
   Polygon[α@*ηx@*σ4x@{{-1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*ηx@*σ4x@{{1, .5, .52}, {0, .3, .52}, {0, .5, .52}}],
   Polygon[α@*ηx@*σ4x@{{-1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@*ηx@*σ4x@{{1, -.5, .52}, {0, -.3, .52}, {0, -.5, .52}}],
   Polygon[α@{{-1.02, -.5, .52}, {-1.02, -.5, -.52}, {-1.02, -.3, 0}}],
   Polygon[α@{{-1.02, -.5, .52}, {-1.02, .5, .52}, {-1.02, 0, .3}}],
   Polygon[α@{{-1.02`, 0.5`, -0.52`}, {-1.02`, -0.5`, -0.52`}, {-1.02`, 0.`, -0.3`}}],
   Polygon[α@{{-1.02`, 0.52`, 0.5`}, {-1.02`, 0.52`, -0.5`}, {-1.02`, 0.3`, 0.`}}],
   Polygon[α@*ρ1@{{-1.02, -.5, .52}, {-1.02, -.5, -.52}, {-1.02, -.3, 0}}],
   Polygon[α@*ρ1@{{-1.02, -.5, .52}, {-1.02, .5, .52}, {-1.02, 0, .3}}], Polygon[
    α@*ρ1@{{-1.02`, 0.5`, -0.52`}, {-1.02`, -0.5`, -0.52`}, {-1.02`, 0.`, -0.3`}}],
   Polygon[α@*ρ1@{{-1.02`, 0.52`, 0.5`}, {-1.02`, 0.52`, -0.5`}, {-1.02`, 0.3`, 0.`}}]
```

```
  },

  {Brown, EdgeForm[],
   Polygon[α@{{.8, -.52, 0}, {0, -.52, .3}, {-.8, -.52, 0}, {0, -.52, -.3}}]},
  {Yellow, Polygon[α@{{-.8, 0, -.52}, {0, .3, -.52}, {.8, 0, -.52}, {0, -.3, -.52}}]},
  {Green, Polygon[α@{{.8, .52, 0}, {0, .52, .3}, {-.8, .52, 0}, {0, .52, -.3}}]},
  {Pink, Polygon[α@{{1.02, -.3, .3}, {1.02, .3, .3}, {1.02, .3, -.3}, {1.02, -.3, -.3}}]},
  {Cyan,
   Polygon[α@{{-1.02, -.3, .3}, {-1.02, .3, .3}, {-1.02, .3, -.3}, {-1.02, -.3, -.3}}]}}
```

In[16]:=

```
ι := TranslationTransform[{0, 0, 0}]
τbx := TranslationTransform[{2, 0, 0}]
τby := TranslationTransform[{0, 1, 0}]
τbz := TranslationTransform[{0, 0, 1}]
τb[v_] := TranslationTransform[v]
ρ1 := TransformationFunction[{{-1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}}]
ρ2 := TransformationFunction[{{1, 0, 0, 0}, {0, -1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}}]
ρ3 := TransformationFunction[{{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, -1, 0}, {0, 0, 0, 1}}]
ξ := ρ1@*ρ2@*ρ3
ξb[o_] := τb[o]@*ξ@*τb[-o]
InverseTF[λ_] := TransformationFunction[Inverse[TransformationMatrix[λ]]]
orthPart[α_] := Take[TransformationMatrix[α], 3, 3]
rotationTF3D[ang_, axis_] := Module[{ta, wab},
  ta = TranslationTransform[axis[[1]]];
  wab = RotationTransform[ang, InverseTF[ta]@axis[[2]]];
  ta@*wab@*InverseTF[ta]]
σ4 := rotationTF3D[Pi / 2, {{-1., 0, 0}, {0, 0, 1}}]
reflectionTF3D[{o_, p_, q_}] := Module[{v},
  v = Cross[p - o, q - o];
  getTF3D[{o, p, q, o + v}, {o, p, q, o - v}]]
glideReflectionTF3D[{p_, q_, r_, t_}] :=
TranslationTransform[t - p]@*reflectionTF3D[{p, q, r}]
rotationTF3D[ang_, axis_] := Module[{ta, wab},
  ta = TranslationTransform[axis[[1]]];
  wab = RotationTransform[ang, InverseTF[ta]@axis[[2]]];
  ta@*wab@*InverseTF[ta]]
rotaryReflectionTF3D[ang_, {o_, p_, q_}] :=
 rotationTF3D[ang, {o, o + Cross[p - o, q - o]}]@*reflectionTF3D[{o, p, q}]
screwDisplacementTF3D[ang_, {u_, v_}] :=
 TranslationTransform[v - u]@*rotationTF3D[ang, {u, v}]
σ4x := rotationTF3D[Pi / 2, {{0, 0, 0}, {1, 0, 0}}]
σ4y := rotationTF3D[Pi / 2, {{0, 0, 0}, {0, 1, 0}}]
σ4z := rotationTF3D[Pi / 2, {{0, 0, 0}, {0, 0, 1}}]
ηx := rotationTF3D[Pi, {{0, 0, 0}, {1, 0, 0}}]
ηy := rotationTF3D[Pi, {{0, 0, 0}, {0, 1, 0}}]
ηz := rotationTF3D[Pi, {{0, 0, 0}, {0, 0, 1}}]
```

In[41]:=
```
classTF3D[δ_] := Module[{M, M1, M2, ordM1, K, tp, rr, gp, det},
  M = TransformationMatrix[δ];
  If[ Norm[M - IdentityMatrix[4]] < .001, Return["ι"]];
  M1 = Take[M, 3, 3];
  If[Norm[M1 - IdentityMatrix[3]] < .001 , Return["T"]];
  If[Norm[M1.M1 - IdentityMatrix[3]] < .001,
   (*Start involutions*)
   K = TransformationMatrix[δ@*δ];
   If[Det[M1] < 0 && Norm[K - IdentityMatrix[4]] > .001, Return["GRF"]];
   If[Det[M1] > 0 && Norm[K - IdentityMatrix[4]] > .001, Return["SD2"]];
   If[Norm[K - IdentityMatrix[4]] < .001 && Det[M1] > 0, Return["H"]];
   If[Norm[K - IdentityMatrix[4]] < .001 && Det[M1] < 0 &&
     Norm[M1 + IdentityMatrix[3]] < .001, Return["INVO"], Return["RF"]];
   Return["U"]];
  (*End Involutions*)
  ordM1 = 1;
  M2 = M1;
  tp = {.3924, -.1245, .7738};
  While[ordM1 < 7 && Norm[M2 - IdentityMatrix[3]] > .005, M2 = M2.M1;
   ordM1++];
  rr = {tp, δ@tp, δ@*δ@tp, δ@*δ@*δ@tp};
  gp = generalPosition3DQ[rr];
  det = Det[M1];
  If[gp,
   If[det > 0,
    Return[Switch[ordM1, 2, "SD2", 3, "SD3", 4, "SD4", 5, "TRSD", 6, "SD6", _, "SDU"]]];
   Return[Switch[ordM1, 4, "RRF4", 6, "RRF6", _, "RRFU"]]];
  Return[Switch[ordM1, 3, "R3", 4, "R4", 5, "R5", 6, "R6", _, "RU"]];
  Return["U1"]]
```

In[42]:=
```
TasTF[S_, tas_] := Module[{len, t, i}, len = Length[S];
  t = tas[S[[len]]];
  i = 1;
  While[i < len, t = t@*tas[S[[len - i]]]; i++];
  t]
```

In[43]:=
```
Options[nFindKeys] = {tol → 1.*^-5};
nFindKeys[A_, v_, OptionsPattern[]] := Module[{S, R},
  S = Normal[A];
  R = Reap[Do[If[Norm[s[[2]] - v] < OptionValue[tol], Sow[s[[1]]]], {s, S}]];
  If[Length[R[[2]]] > 0, Return[R[[2, 1]]], Return[{{1}}]]]
```

In[45]:=
```
Options[transGroup] = {tol → .001};
transGroup[tas_, tp_, n_, OptionsPattern[]] := Module[{k, A, An, V, K},
  k = Length[tas];
  A = <| ({#} → tas[#] @tp) & /@ Range[k] |>;
  Do[An = <| (# → TasTF[#, tas] @tp) & /@ Tuples[Range[k], j] |>;
   A = Join[A, An], {j, 2, n}];
  V = DeleteDuplicates[Values[A], Norm[#1 - #2] < OptionValue[tol] &];
  K = Table[First[nFindKeys[A, V[[i]]]], {i, Length[V]}];
  Echo[Length[V], "number of group elements calculated "];
  K]
```

In[47]:=
```
GroupFindKey[g_, G_, tas_, tp_] := Module[{S, R, gv},
  S = Table[α → TasTF[α, tas] @tp, {α, G}];
  gv = g@tp;
  R = Reap[Do[If[Norm[s[[2]] - gv] < 1.*^-5, Sow[s[[1]]]], {s, S}]];
  If[Length[R[[2]]] > 0, Return[R[[2, 1]][[1]]], Return[Fail]]]
```