1a. Definitions of group and its operations

See 2.1.

Many of the routines defined here are use throughout this notebook.

Routine **inG** checks whether a matrix is an element of G=SU(2,1)

In[ ]:=
```
(I21 = {{1, 0, 0}, {0, 1, 0}, {0, 0, -1}}) // MatrixForm;
Clear[inG]
inG[g_] := ((Det[g] == 1) && Simplify[Conjugate[Transpose[g]].I21.g == I21]) //. gensub
gensub = {zz_ Conjugate[zz_] :> Abs[zz]^2, zz_^ee_ Conjugate[zz_]^ee_ :> Abs[zz]^(2 ee),
    Conjugate[pp_ qq_] :> Conjugate[pp] Conjugate[qq],
    Conjugate[pp_ + qq_] :> Conjugate[pp] + Conjugate[qq],
    Abs[pp_ qq_] :> Abs[pp] Abs[qq], Abs[pp_ / qq_] :> Abs[pp] / Abs[qq],
    Abs[pp_ + qq_]^2 :> Abs[pp]^2 + Abs[qq]^2 + pp Conjugate[qq] + Conjugate[pp] qq};
```

The special elements can be used in two ways: as matrices, with notation ending in m; and as elements for formal manipulation, with notation ending in s.

In[ ]:=
```
Clear[nm, ns, am, as, mm, ms, km, ks, wm, ws, hm, hs]
nm[b_, r_] := {{1 + I r - Abs[b]^2 / 2, b, -I r + Abs[b]^2 / 2},
   {-Conjugate[b], 1, Conjugate[b]}, {I r - Abs[b]^2 / 2, b, 1 - I r + Abs[b]^2 / 2}};
nm[x_, y_, r_] := nm[x + I (y), r] /. {Conjugate[x] -> x,
     Conjugate[y] -> y, Abs[x + I y]^2 -> x^2 + y^2} // Simplify
mm[z_] = DiagonalMatrix[{z, z^(-2), z}];
am[y_] = {{(y + 1 / y) / 2, 0, (y - 1 / y) / 2}, {0, 1, 0}, {(y - 1 / y) / 2, 0, (y + 1 / y) / 2}};
km[zt_, al_, bt_] := {{zt al, zt bt, 0}, {-zt Conjugate[bt], zt Conjugate[al], 0},
   {0, 0, zt^(-2)}};
wm = km[1, -1, 0];
hm[c_] = am[Abs[c]].mm[c / Abs[c]] // Simplify;
```

For the matrices we check whether they are in G.

In[ ]:=
```
Clear[b, r, x, y]
nm[b, r] // MatrixForm
inG[%] /. Conjugate[r] → r
nm[x, y, r] == nm[x + I y, r] // ComplexExpand
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 + i\, r - \frac{\text{Abs}[b]^2}{2} & b & -i\, r + \frac{\text{Abs}[b]^2}{2} \\ -\text{Conjugate}[b] & 1 & \text{Conjugate}[b] \\ i\, r - \frac{\text{Abs}[b]^2}{2} & b & 1 - i\, r + \frac{\text{Abs}[b]^2}{2} \end{pmatrix}$$

Out[ ]= True

Out[ ]= True

In[ ]:=
```
Clear[t]
am[t] // MatrixForm
inG[%] /. Conjugate[t] → t /. Abs[t] → t
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} \frac{1}{2}\left(\frac{1}{t} + t\right) & 0 & \frac{1}{2}\left(-\frac{1}{t} + t\right) \\ 0 & 1 & 0 \\ \frac{1}{2}\left(-\frac{1}{t} + t\right) & 0 & \frac{1}{2}\left(\frac{1}{t} + t\right) \end{pmatrix}$$

Out[ ]= True

In[ ]:=
```
Clear[zt]
mm[zt] // MatrixForm
inG[mm[zt]] /. Abs[zt] → 1
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} zt & 0 & 0 \\ 0 & \frac{1}{zt^2} & 0 \\ 0 & 0 & zt \end{pmatrix}$$

Out[ ]= True

In[ ]:=
```
Clear[zt, al, bt]
km[zt, al, bt] // MatrixForm
inG[km[zt, al, bt]] /. Conjugate[zt] → zt ^ (-1) /. Abs[zt] → 1 /.
   Abs[bt] ^ 2 → 1 - Abs[al] ^ 2 // Simplify
mm[zt] == km[zt ^ (-1 / 2), zt ^ (3 / 2), 0] /. Conjugate[zt^(3/2)] → zt ^ (-3 / 2)
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} \text{al zt} & \text{bt zt} & 0 \\ -\text{zt Conjugate}[\text{bt}] & \text{zt Conjugate}[\text{al}] & 0 \\ 0 & 0 & \frac{1}{zt^2} \end{pmatrix}$$

Out[ ]= True

Out[ ]= True

The substitution rule **Gsub** gives the group operations on the symbolic elements

In[ ]:=
```
Clear[id, iv]
Gsub = {ns[b_, r_] ** ns[b1_, r1_] :> ns[b + b1, r + r1 + Im[Conjugate[b] b1]],
    ns[x_, y_, r_] ** ns[x1_, y1_, r1_] :> ns[x + x1, y + y1, r + r1 + x y1 - y x1], ns[0, 0, 0] → id,
    ns[0, 0] → id, iv[ns[b_, r_]] :> ns[-b, -r], iv[ns[x_, y_, r_]] :> ns[-x, -y, -r],
    as[t_] ** as[t1_] :> as[t t1],
    as[1] → id,
    iv[as[t_]] :> as[1 / t],
    ms[zt_] ** ms[zt1_] :> ms[zt zt1],
    ms[1] → id, iv[ms[zt_]] :> ms[1 / zt],
    as[t_] ** ns[b_, r_] :> ns[t b, t ^ 2 r] ** as[t],
    ms[zt_] ** ns[b_, r_] :> ns[zt ^ 3 b, r] ** ms[zt],
    as[t_] ** ns[x_, y_, r_] :> ns[t x, t y, t ^ 2 r] ** as[t],
    hs[c_] :> as[Abs[c]] ** ms[c / Abs[c]], aa_ ** id :> aa, id ** aa_ :> aa,
    ms[zt_] ** as[t] :> as[t] ** ms[zt], iv[pp_ ** qq_] :> iv[qq] ** iv[pp]};
```

Some checks of the group operations

In[ ]:=
```
Clear[b, r, b1, r1, x, y, x1, y1]
ns[b, r].ns[b1, r1] == ns[b + b1, r + r1 + Im[Conjugate[b] b1]] /. ns → nm /.
    Im[zz_] :> (zz - Conjugate[zz]) / (2 I) //. gensub // Simplify
% /. Abs[zz_ + zz1_] ^ 2 :> Abs[zz] ^ 2 + Abs[zz1] ^ 2 + zz Conjugate[zz1] + Conjugate[zz] zz1
ns[x, y, r].ns[x1, y1, r1] == ns[x + x1, y + y1, r + r1 - y x1 + x y1] /. ns → nm // Simplify
```

Out[ ]= True

Out[ ]= True

Out[ ]= True

In[ ]:=
```
Clear[b, r]
iv[ns[b, r]] == ns[-b, -r] /. iv → Inverse /. ns → nm //. gensub
ns[b, r].iv[ns[b, r]] /. iv → Inverse /. ns → nm //. gensub // Simplify // MatrixForm
```

Out[ ]= True

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In[ ]:=
```
Clear[t, zt, b, r]
as[t].ms[zt].ns[b, r] == ns[zt ^ 3 t b, t ^ 2 r].as[t].ms[zt] /. {as → am, ns → nm, ms → mm} //.
    gensub /. {Conjugate[t] → t, Conjugate[zt] → 1 / zt, Abs[zt] → 1, Abs[t] → t} // Simplify
```

Out[ ]= True