# Experimenting with Pauli matrices to implement GA(3,0)

```
ClearAll[e1, e2, e3, e12, e23, e31, e123, q, r, s, t, a, b,
 c, p, labels, m, grade01, grade23, grade0, grade1, grade2, grade3]
$Assumptions = {q, r, s, t, a, b, c, p} > 0;
one = IdentityMatrix[2];
e1 = PauliMatrix[1] ;
e2 = PauliMatrix[2] ;
e3 = PauliMatrix[3] ;
e12 = e1 . e2 ;
e23 = e2 . e3 ;
e31 = e3 . e1 ;
e123 = e1 . e2 . e3 ;

m0 = q one ;
m1 = r e1 + s e2 + t e3 ;
m2 = a e12 + b e23 + c e31 ;
m3 = p e123 ;
m = m0 + m1 + m2 + m3 ;

labels = {
   {"1", one},
   {"σ1", e1},
   {"σ2", e2},
   {"σ3", e3},
   {"σ12", e12},
   {"σ23", e23},
   {"σ31", e31},
   {"σ123", e123} } ;

(Row[{# // First, " = ", # // Last // MatrixForm }] & /@ labels) // Column

grade0 := one (# / 2 // Tr // Re // Simplify) &;
grade3 := I one (# / 2 // Tr // Im // Simplify) &;
grade01 := (((# + (# // ConjugateTranspose)) / 2) // Simplify) &;
grade23 := (((# - (# // ConjugateTranspose)) / 2) // Simplify) &;
grade1 := ((grade01[#] - grade0[#]) // Simplify) &;
grade2 := ((grade23[#] - grade3[#]) // Simplify) &;


m // MatrixForm
```

```
m0 // MatrixForm
m1 // MatrixForm
m2 // MatrixForm
m3 // MatrixForm

(*(m0 - (m//grade0))//MatrixForm
(m1 -(m//grade1))//MatrixForm
    (m2 -(m//grade2))//MatrixForm
(m3 -(m//grade3))//MatrixForm*)
```

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

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

$$\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\sigma_{12} = \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}$$

$$\sigma_{23} = \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix}$$

$$\sigma_{31} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

$$\sigma_{123} = \begin{pmatrix} i & 0 \\ 0 & i \end{pmatrix}$$

$$\begin{pmatrix} i\,a + i\,p + q + t & i\,b + c + r - i\,s \\ i\,b - c + r + i\,s & -i\,a + i\,p + q - t \end{pmatrix}$$

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

$$\begin{pmatrix} t & r - i\,s \\ r + i\,s & -t \end{pmatrix}$$

$$\begin{pmatrix} i\,a & i\,b + c \\ i\,b - c & -i\,a \end{pmatrix}$$

$$\begin{pmatrix} i\,p & 0 \\ 0 & i\,p \end{pmatrix}$$

```
(*ClearAll[scalarType, vectorType, bivectorType,
   trivectorType, multivectorType, Scalar, Vector, Bivector,
   Trivector, pScalarQ, pVectorQ, pBivectorQ, pTrivectorQ]
  {scalarType, vectorType, bivectorType, trivectorType, multivectorType} =
 Range[5];
Scalar[v_] := {scalarType,v IdentityMatrix[2]};
Vector[v_, k_Integer /; k > 0 && k < 4] := {vectorType, v PauliMatrix[k] };
Bivector[v_, k_Integer /; k > 0 && k < 4, j_Integer  /; j > 0 && j < 4] :=
 {bivectorType, v PauliMatrix[k] . PauliMatrix[j] };
Trivector[v_] := {trivectorType, v I IdentityMatrix[2]};


pScalarQ[m:{scalarType, _}] := True ;
pScalarQ[m:{_Integer, _}] := False ;
pVectorQ[m:{vectorType, _}] := True ;
pVectorQ[m:{_Integer, _}] := False ;
pBivectorQ[m:{bivectorType, _}] := True ;
pBivectorQ[m:{_Integer, _}] := False ;
pTrivectorQ[m:{trivectorType, _}] := True ;
pTrivectorQ[m:{_Integer, _}] := False ;


ClearAll[DotProduct]
  DotProduct[ v1_?pVectorQ, v2_?pVectorQ] :=
 Module[{a = (v1//Last),b = (v2//Last)},{scalarType, (a. b + b.a)/2}]

    DotProduct[ Vector[1, 1], Vector[1, 1]] // Last // MatrixForm
   DotProduct[ Vector[1, 1],  Bivector[1, 1]]
   WedgeP
*)
```

```
(2 IdentityMatrix[2])[[1, 1]]
```

```
2
```