> $restart; assume(a,'positive'); with(LinearAlgebra);$

[&x, Add, Adjoint, BackwardSubstitute, BandMatrix, Basis, BezoutMatrix, BidiagonalForm, BilinearForm, CARE, CharacteristicMatrix, CharacteristicPolynomial, Column, ColumnDimension, ColumnOperation, ColumnSpace, CompanionMatrix, CompressedSparseForm, ConditionNumber, ConstantMatrix, ConstantVector, Copy, CreatePermutation, CrossProduct, DARE, DeleteColumn, DeleteRow, Determinant, Diagonal, DiagonalMatrix, Dimension, Dimensions, DotProduct, EigenConditionNumbers, Eigenvalues, Eigenvectors, Equal, ForwardSubstitute, FrobeniusForm, FromCompressedSparseForm, FromSplitForm, GaussianElimination, GenerateEquations, GenerateMatrix, Generic, GetResultDataType, GetResultShape, GivensRotationMatrix, GramSchmidt, HankelMatrix, HermiteForm, HermitianTranspose, HessenbergForm, HilbertMatrix, HouseholderMatrix, IdentityMatrix, IntersectionBasis, IsDefinite, IsOrthogonal, IsSimilar, IsUnitary, JordanBlockMatrix, JordanForm, KroneckerProduct, LA_Main, LUDecomposition, LeastSquares, LinearSolve, LyapunovSolve, Map, Map2, MatrixAdd, MatrixExponential, MatrixFunction, MatrixInverse, MatrixMatrixMultiply, MatrixNorm, MatrixPower, MatrixScalarMultiply, MatrixVectorMultiply, MinimalPolynomial, Minor, Modular, Multiply, NoUserValue, Norm, Normalize, NullSpace, OuterProductMatrix, Permanent, Pivot, PopovForm, ProjectionMatrix, QRDecomposition, RandomMatrix, RandomVector, Rank, RationalCanonicalForm, ReducedRowEchelonForm, Row, RowDimension, RowOperation, RowSpace, ScalarMatrix, ScalarMultiply, ScalarVector, SchurForm, SingularValues, SmithForm, SplitForm, StronglyConnectedBlocks, SubMatrix, SubVector, SumBasis, SylvesterMatrix, SylvesterSolve, ToeplitzMatrix, Trace, Transpose, TridiagonalForm, UnitVector, VandermondeMatrix, VectorAdd, VectorAngle, VectorMatrixMultiply, VectorNorm, VectorScalarMultiply, ZeroMatrix, ZeroVector, Zip] **(1)**

Define lattice translation

> $T1 := Vector(3, [a, 0, 0]); T2 := Vector(3, [0, a, 0]); T3 := Vector(3, [0, 0, a]);$

$$T1 := \begin{bmatrix} a\sim \\ 0 \\ 0 \end{bmatrix}$$

$$T2 := \begin{bmatrix} 0 \\ a\sim \\ 0 \end{bmatrix}$$

(2)

$$T3 := \begin{bmatrix} 0 \\ 0 \\ a\sim \end{bmatrix} \quad (2)$$

Define reciprocal lattice translation

> $G1 := Vector\left(3, \left[\frac{2\cdot \text{Pi}}{a}, 0, 0\right]\right); G2 := Vector\left(3, \left[0, \frac{2\cdot \text{Pi}}{a}, 0\right]\right);$
> $G3 := Vector\left(3, \left[0, 0, \frac{2\cdot \text{Pi}}{a}\right]\right);$

$$G1 := \begin{bmatrix} \frac{2\,\pi}{a\sim} \\ 0 \\ 0 \end{bmatrix}$$

$$G2 := \begin{bmatrix} 0 \\ \frac{2\,\pi}{a\sim} \\ 0 \end{bmatrix}$$

$$G3 := \begin{bmatrix} 0 \\ 0 \\ \frac{2\,\pi}{a\sim} \end{bmatrix} \quad (3)$$

> $\text{tau} := Vector(3, [0.5\cdot a, 0.5\cdot a, 0.5\cdot a]);$

$$\tau := \begin{bmatrix} 0.5\, a\sim \\ 0.5\, a\sim \\ 0.5\, a\sim \end{bmatrix} \quad (4)$$

>

> $\eta := \frac{4}{a^2}; \Omega := a^3; con1 := \frac{4\cdot \text{Pi}}{\text{Omega}}; con2 := \text{sqrt}\left(\frac{\eta}{\text{Pi}}\right);$

$$\eta := \frac{4}{a\sim^2}$$

$$\Omega := a\sim^3$$

$$con1 := \frac{4\,\pi}{a\sim^3}$$

(5)

$$con2 := \frac{2}{a\sim\sqrt{\pi}} \tag{5}$$

Initial terms -- Cl−Cl **and** *Cs−Cs*

\>

\> $tot := -evalf(con2 \cdot 2);$

$$tot := -\frac{2.256758334}{a\sim} \tag{6}$$

\> **for** n **from** -8 **by** 1 **while** $n < 8$ **do** **for** m **from** -8 **by** 1 **while** $m < 8$ **do** **for** l **from** -8 **by** 1 ***while*** $\boldsymbol{l < 8}$ ***do*** **if** $(\boldsymbol{n \neq 0}$ **or** $\boldsymbol{m \neq 0}$ **or** $\boldsymbol{l \neq 0})$ **then** $\boldsymbol{g} :=$ $(\boldsymbol{n \cdot G1 + m \cdot G2 + l \cdot G3})$; $\boldsymbol{tot := tot} + evalf\left(\boldsymbol{2 \cdot con1 \cdot (1 - \exp(-I \cdot DotProduct(g, \tau)))} \cdot \frac{\exp\left(-\frac{DotProduct(g, g)}{\text{eta}}\right)}{DotProduct(g, g)}\right)$ **end if end do end do end do**; $evalf(tot)$;

$$-\frac{2.256758334}{a\sim} + \frac{0.0003951360355 + 1.907345200 \times 10^{-24}\,\mathrm{I}}{a\sim} \tag{7}$$

\> **for** n **from** -8 **by** 1 **while** $n < 8$ **do** **for** m **from** -8 **by** 1 **while** $m < 8$ **do** **for** l **from** -8 **by** 1 ***while*** $\boldsymbol{l < 8}$ ***do*** $\boldsymbol{t := (n \cdot T1 + m \cdot T2 + l \cdot T3)}$; $\boldsymbol{tot := tot} - evalf\left(\frac{\boldsymbol{2 \cdot \left(erfc\left(\frac{\text{sqrt}(eta)}{2} \cdot VectorNorm(\tau + t, 2)\right)\right)}}{VectorNorm(\text{tau} + t, 2)}\right)$; **if** $(n \neq 0$ **or** $m \neq 0$ **or** $l \neq 0)$ **then** $\boldsymbol{tot := tot} + evalf\left(\frac{\boldsymbol{2 \cdot erfc\left(\frac{sqrt(eta)}{2} \cdot VectorNorm(t, 2)\right)}}{\boldsymbol{VectorNorm(t, 2)}}\right)$ **end if end do end do end do**; $evalf(tot)$;

$$-\frac{4.071118106}{a\sim} + \frac{0.0003951360355 + 1.907345200 \times 10^{-24}\,\mathrm{I}}{a\sim} \tag{8}$$

```
> Re(%);
```

$$-\frac{4.070722970}{a\sim} \tag{9}$$

```
>
```