

MatrixSparse TODENSE Example

```

Function DemoMatrixSparseTODENSE()
    // Create the Wikipedia example 4x4 matrix in COO format
    Make/FREE/D values = {5, 8, 3, 6}
    Make/FREE/L rows = {1, 1, 2, 3}
    Make/FREE/L columns = {0, 1, 2, 1}

    // Create a dense matrix from the sparse matrix
    MatrixSparse rowsA=4, colsA=4, cooA={values,rows,columns}, operation=TODENSE
    WAVE M_cooToDense           // Output from TODENSE

    // Print the dense output matrix
    Print M_cooToDense
End

```

MatrixSparse TRSV

TRSV solves a system of linear equations for the triangular sparse matrix A. Symbolically it solves for output matrix M_TRSVOut where:

`smA*M_TRSVOut = alpha*vX`

Inputs: Sparse matrix A in CSR format, alpha, vector X.

Output: Dense matrix M_TRSVOut.

MatrixSparse TRSV Example

```

// This is based on the example in the Row Reduction section
// of the Wikipedia page
// https://en.wikipedia.org/wiki/System\_of\_linear\_equations
// The system of equations is:
//      x + 3y - 2z = 5
//      3x + 5y + 6z = 7
//      2x + 47 + 3z = 8
// This gives the following augmented matrix:
//      1  3  -2  5
//      3  5   6   7
//      2  4   3   8
// The solution is: x=-15, y=8, z=2
// Because the MatrixSparse TRSV operation requires a triangularized coefficient
// matrix, we start with the upper-triangular version of the augmented matrix
// obtained using Gauss-Jordan elimination:
//      1  3  -2  5
//      0  1  -3  2
//      0  0   1   2
// We create an equivalent sparse matrix using MatrixSparse TOCSR.
// We then create the corresponding solution vector {5, 5, 2}.
// We then call MatrixSparse TRSV and obtain the solution set {-15, 8, 2}
Function DemoMatrixSparseTRSV()
    // Create dense upper-triangular matrix representing coefficients
    Make/FREE/D/N=(3,3) utMat
    utMat[0][0] = {1, 0, 0}          // Column 0
    utMat[0][1] = {3, 1, 0}          // Column 1
    utMat[0][2] = {-2, -3, 1}        // Column 2

    // Create sparse version of upper triangular matrix in CSR format
    MatrixSparse rowsA=3, colsA=3, matrixB=utMat, operation=TOCSR
    WAVE values = W_CSRValues
    WAVE columns = W_CSRColumns
    WAVE ptrB = W_CSRPointerB

```