MatrixSparse MM Example

```
Function DemoMatrixSparseMM()
   // Define sparse matrix in CSR format
   Make/FREE/D values = {5, 8, 3, 6}
   Make/FREE/L columns = {0, 1, 2, 1}
   Make/FREE/L ptrB = {0, 0, 2, 3, 4}

   // Create a dense matrix
   Make/FREE/D matrix = { {1,0,0,0}, {0,1,0,0}, {0,0,1,0}, {0,0,0,1} }

   // Multiply the sparse matrix by the dense matrix
   MatrixSparse rowsA=4, colsA=4, csrA={values,columns,ptrB}, matrixB=matrix,
            operation=MM

   // Create wave reference for output dense matrix
   WAVE M_MMOut         // Output from MV

   Print M_MMOut
End
```

MatrixSparse MV

MV computes the product of a sparse matrix which must be in CSR format and a vector producing an output vector. Symbolically:

```
W_MV = alpha*smA*vX + beta*vY
```

Inputs: alpha, sparse matrix A in CSR format, vector X, and optionally beta and vector Y.

If you leave beta with its default value of 0 by omitting the beta keyword, the beta*vY term is not computed and you do not need to specify the vY input.

Output: Vector W_MV.

MatrixSparse MV Example

```
Function DemoMatrixSparseMV()
   // Define sparse matrix in CSR format
   Make/FREE/D values = {5, 8, 3, 6}
   Make/FREE/L columns = {0, 1, 2, 1}
   Make/FREE/L ptrB = {0, 0, 2, 3, 4}

   // Create a vector
   Make/FREE/D vector = {1, 1, 1, 1}

   // Multiply the sparse matrix by the vector
   MatrixSparse rowsA=4, colsA=4, csrA={values,columns,ptrB}, vectorX=vector,
            operation=MV

   // Create wave reference for output vector
   WAVE W_MV      // Output from MV

   Print W_MV
End
```

MatrixSparse SMSM

SMSM computes the product of a two sparse matrices. Symbolically:

```
smOut = smA * smG
```

Inputs: Sparse matrix A in CSR format, sparse matrix G in CSR format.