APL64 MathNet Interface

Table of Contents

Overview

The APL64 ⎕MathNet system function is an interface to the [MathNet.Numerics.LinearAlgebra](#) **toolkit.**

Syntax

Result←⎕MathNet Action [ActionArgs]

Result: result of the specified action. Refer to ⎕MathNet '?' for a summary of result types

Action: Case insensitive text indicating the user-selected action

ActionArgs optional, depending on the selected action.

For □MathNet mathematical actions with a rank-2 matrix argument:

- A scalar input will be processed as a 1 x 1 matrix
- A vector input will be processed as a (vector length x 1) matrix

For □MathNet mathematical actions with a rank-1 vector argument:

- A scalar input will be processed as a 1-element vector
- A matrix input will be processed as its row-order ravel

Numerical Algorithms

MathNet uses numerical algorithms to obtain results. Since these algorithms are implemented on digital computers, in some cases the results obtained may be approximations of the theoretical mathematical results.

Numerically Derived Matrix Inverse

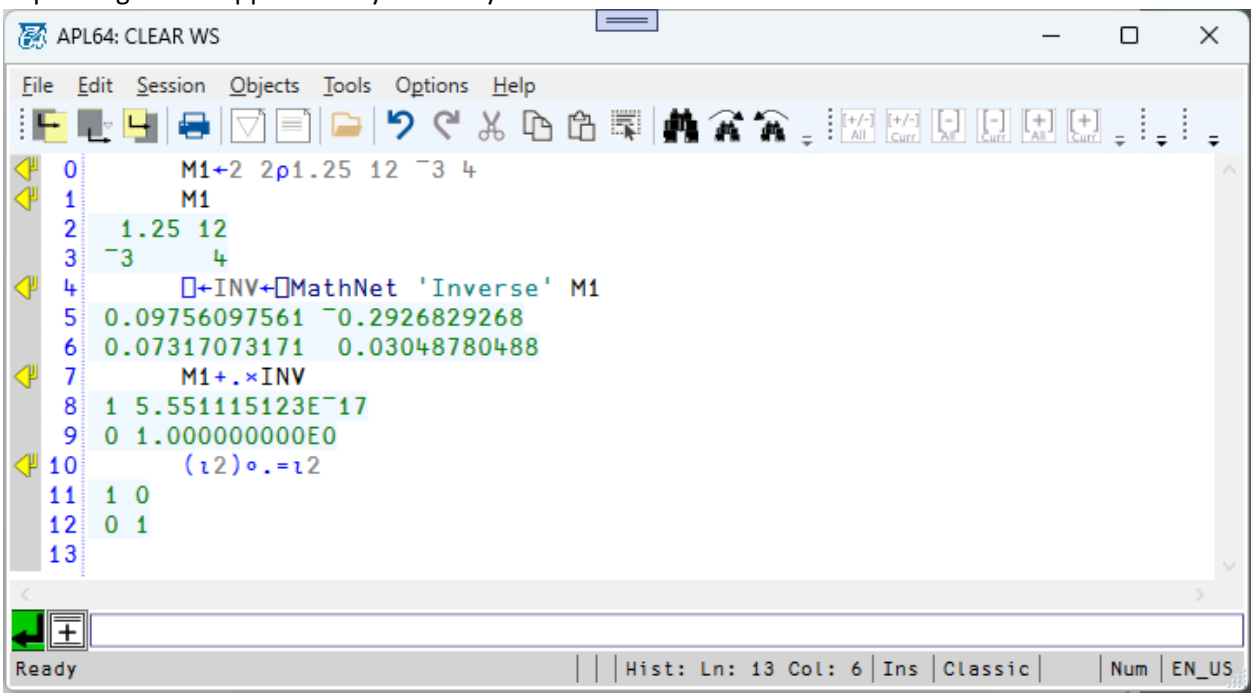
The matrix inverse of an invertible, square matrix with integer elements can have floating point elements. The matrix inverse obtained via MathNet may provide approximate element values. The theoretical mathematical definition of the matrix inverse of the matrix M is the matrix, INV which satisfies:

(M+.×INV) ≡ Identity matrix

(INV+.×M) ≡ Identity matrix

identity matrix is (⍳N)∘.=⍳N←1↑⍴M

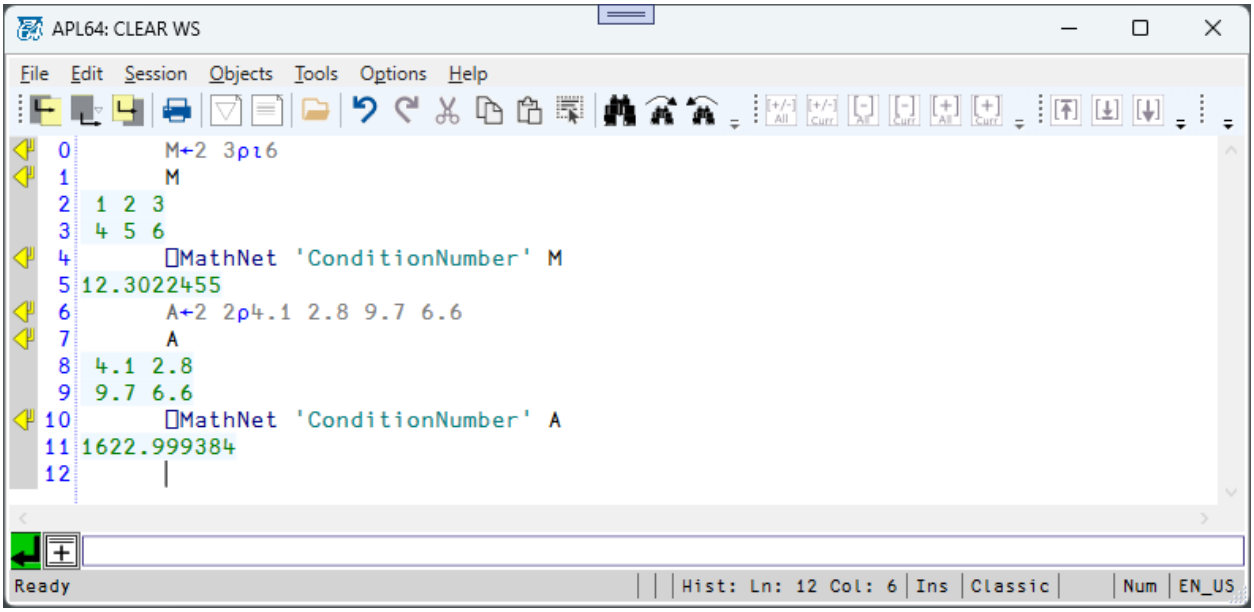
This example illustrates that the MathNet inverse for the matrix M1 is an approximation which depending on the application system may be sufficient.




```
M1←2 2⍴1.25 12 ¯3 4
M1
⎕←INV←⎕MathNet 'Inverse' M1
M1+.×INV
(⍳2)∘.=⍳2
```

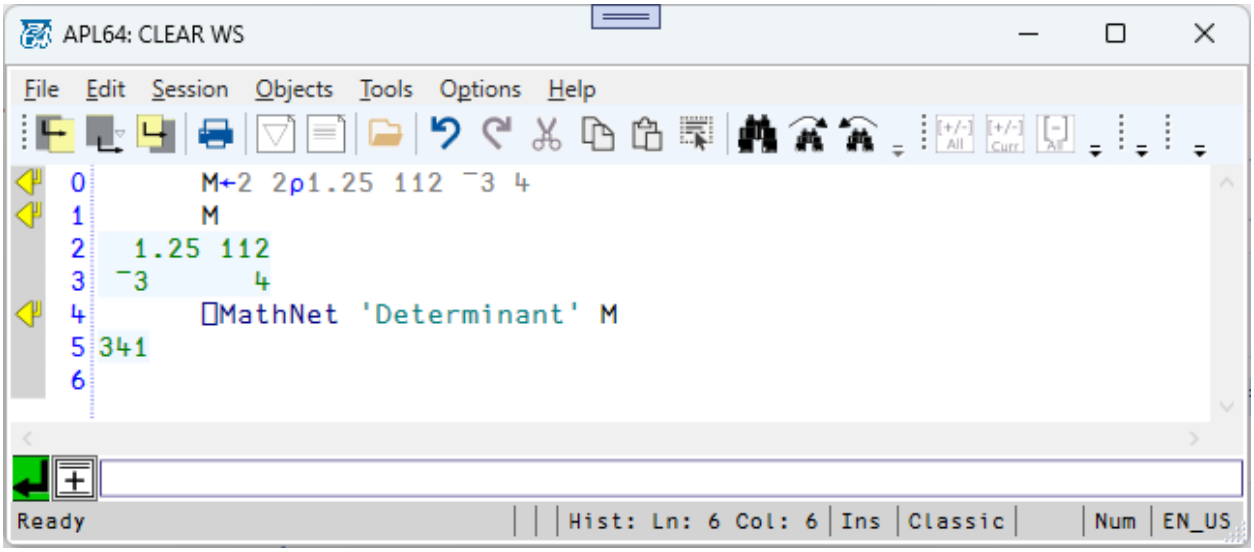
⎕MathNet Actions in APL64:

ConditionNumber



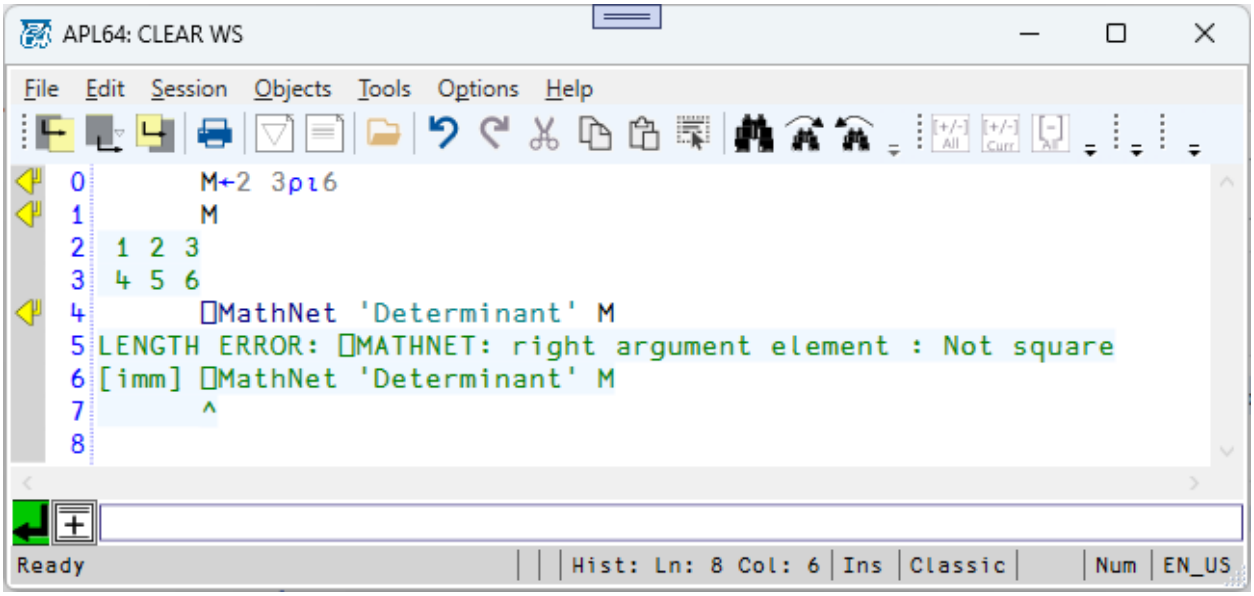
```
M←2 3⍴⍳6
M
⎕MathNet 'ConditionNumber' M
A←2 2⍴4.1 2.8 9.7 6.6
A
⎕MathNet 'ConditionNumber' A
```

Determinant of a square matrix

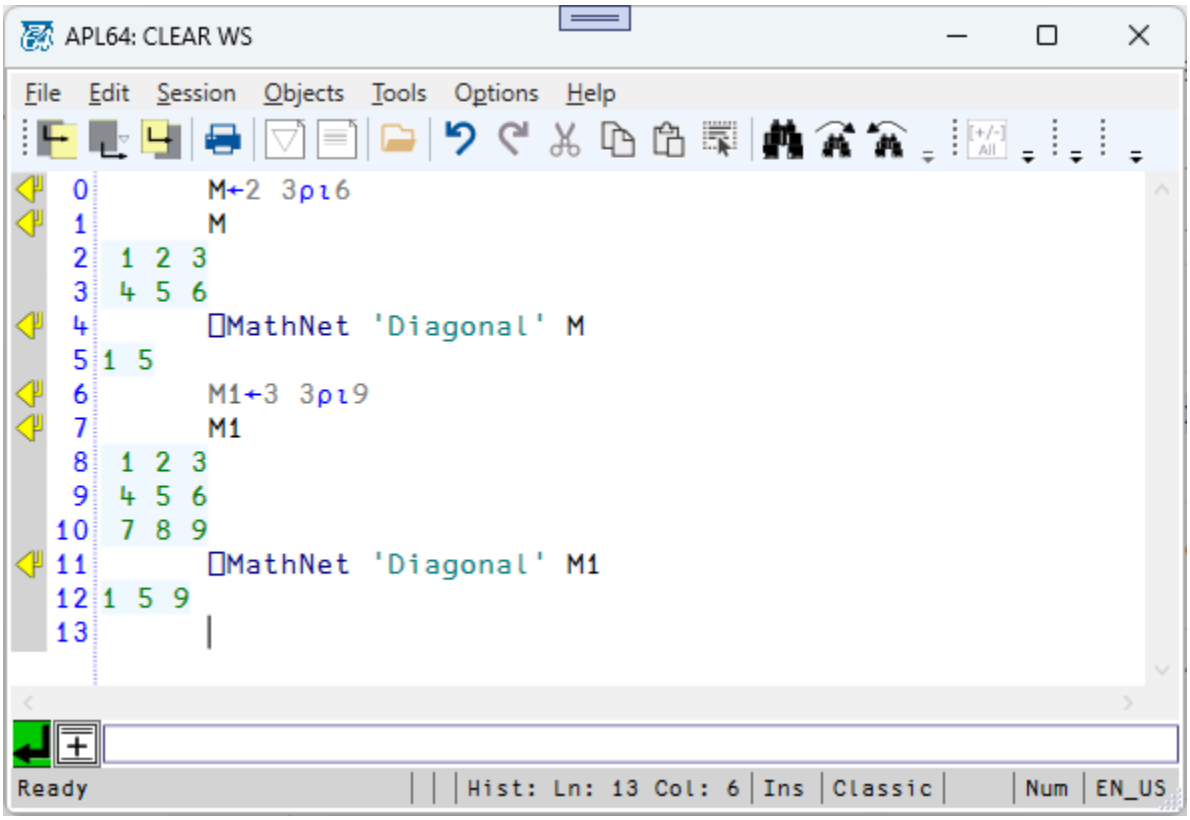



```
M←2 2⍴1.25 112 ¯3 4
M
⎕MathNet 'Determinant' M
```

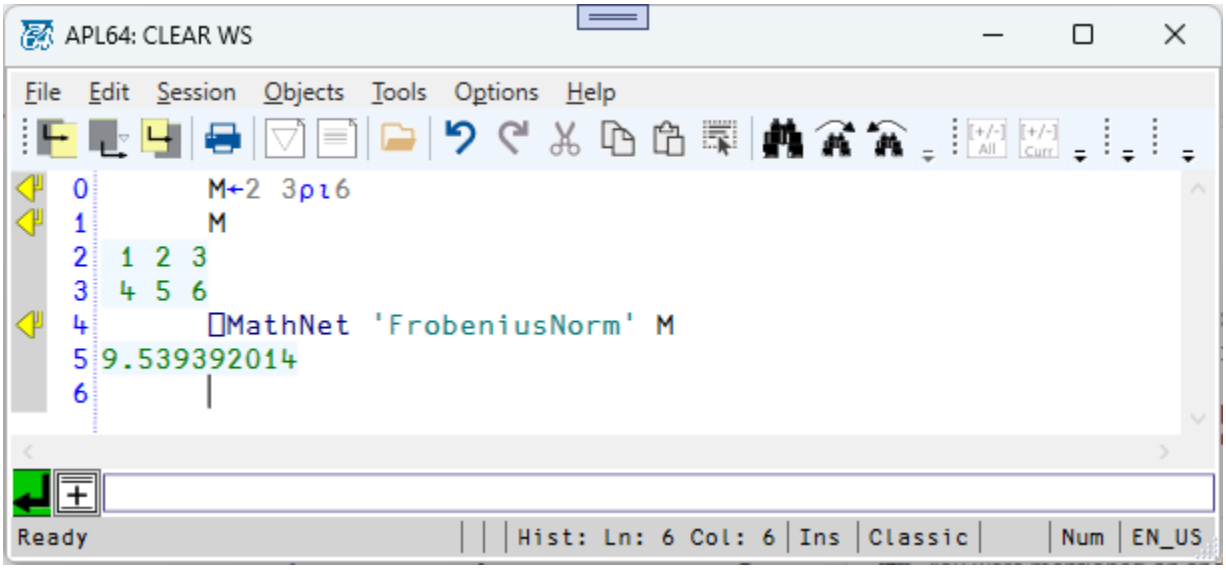
Determinant of non-square matrix does not exist



Diagonal

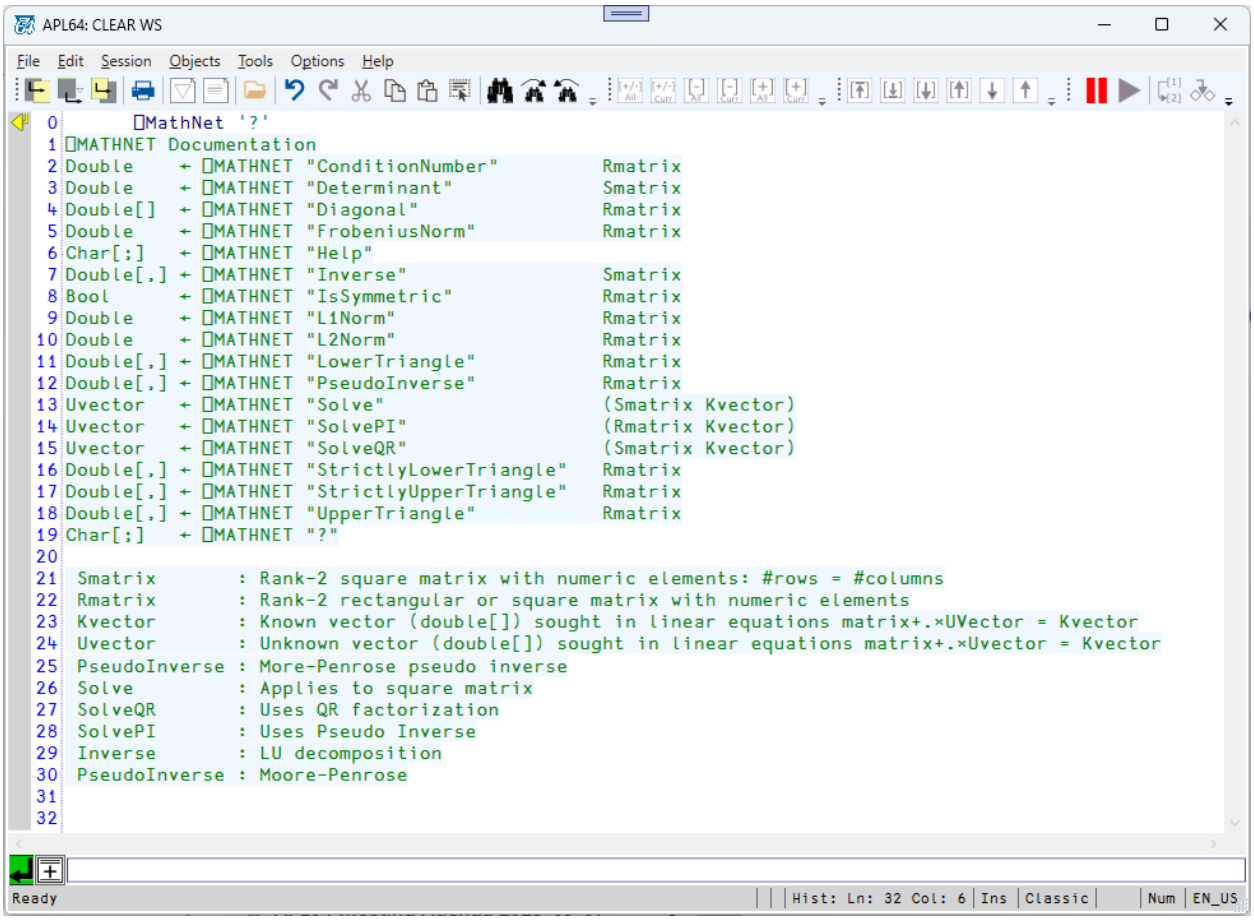


Frobenius Norm



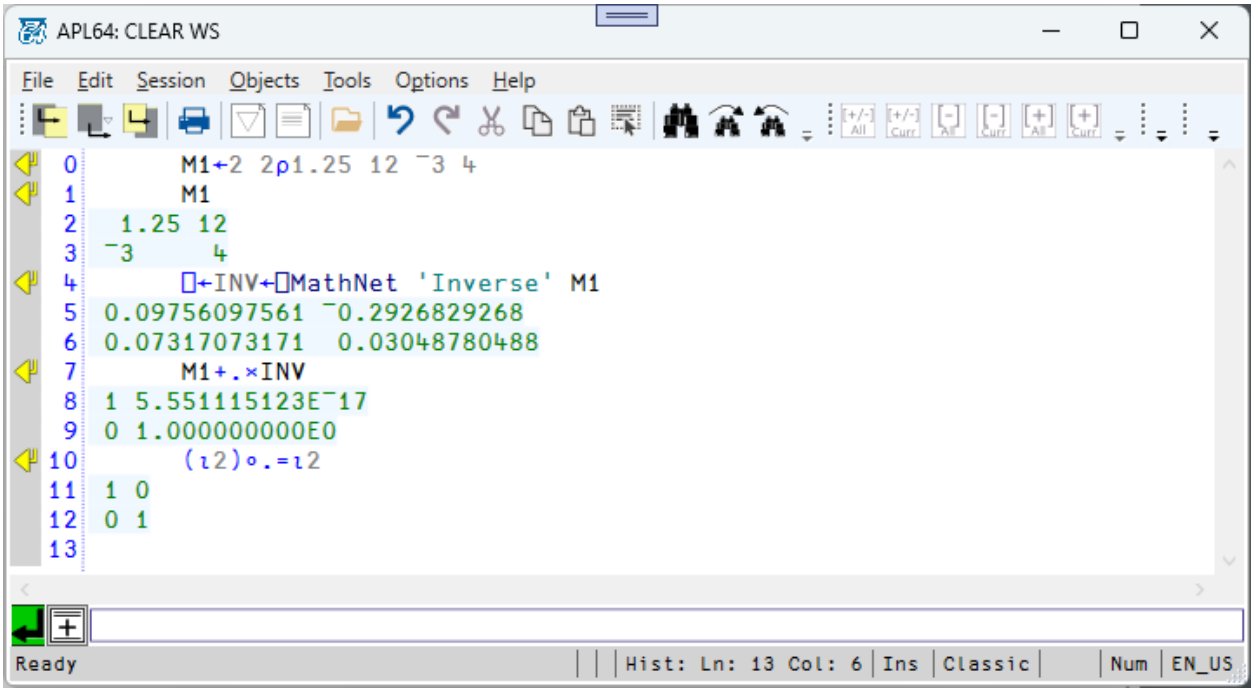
Help

Same as ⎕MathNet '?'



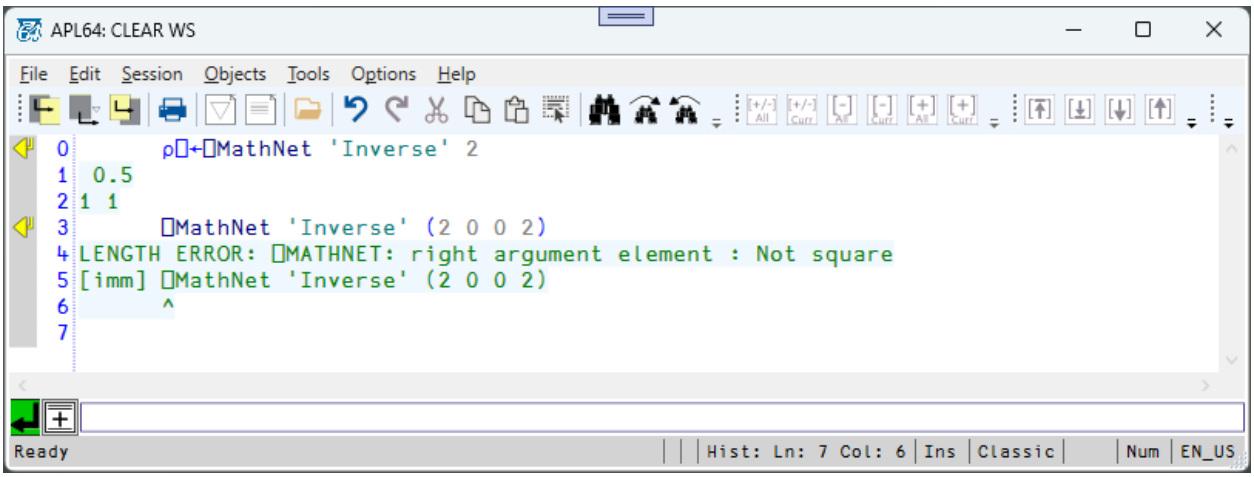
Inverse of a square matrix

In some cases a numerical approximation will be returned.

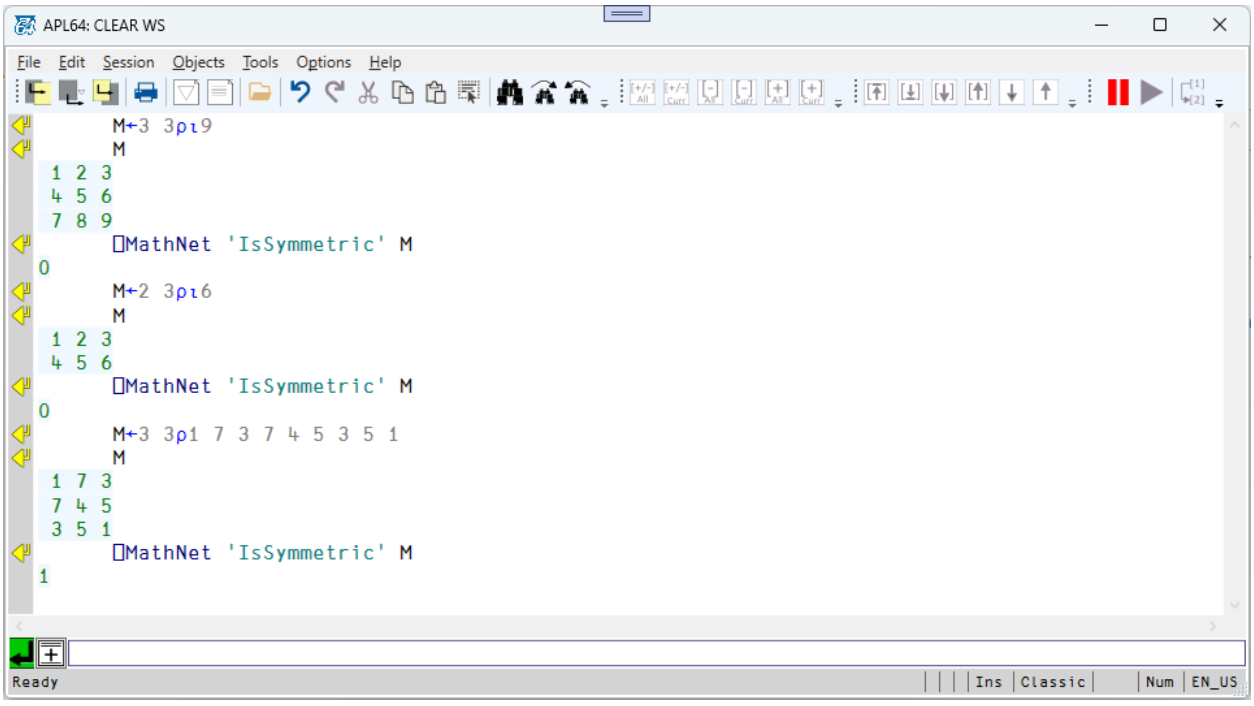


```
M←2 2⍴1.25 12 ¯3 4
M
⎕←INV←⎕MathNet 'Inverse' M1
M1+.×INV
(⍳2)∘.=⍳2
```

Argument must be a square matrix or convertible to a square matrix:

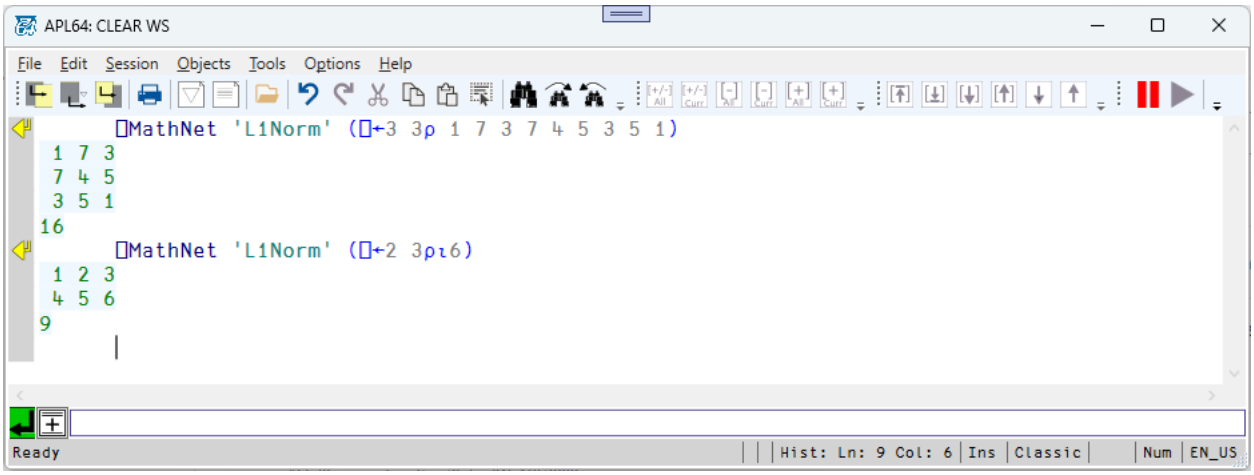


IsSymmetric



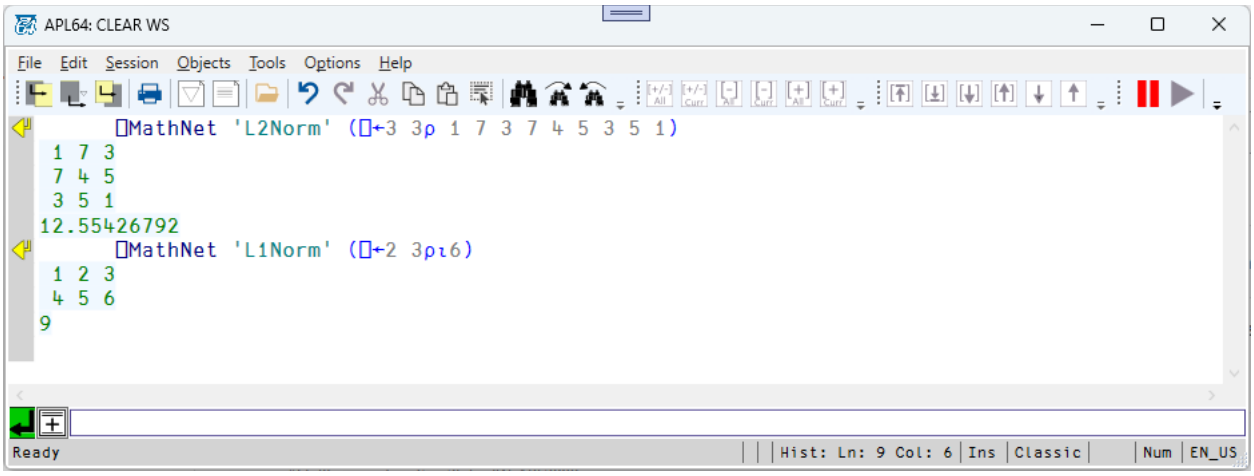

```
M←3 3⍴1 7 3 7 4 5 3 5 1
⎕MathNet 'IsSymmetric' M
```

L1Norm



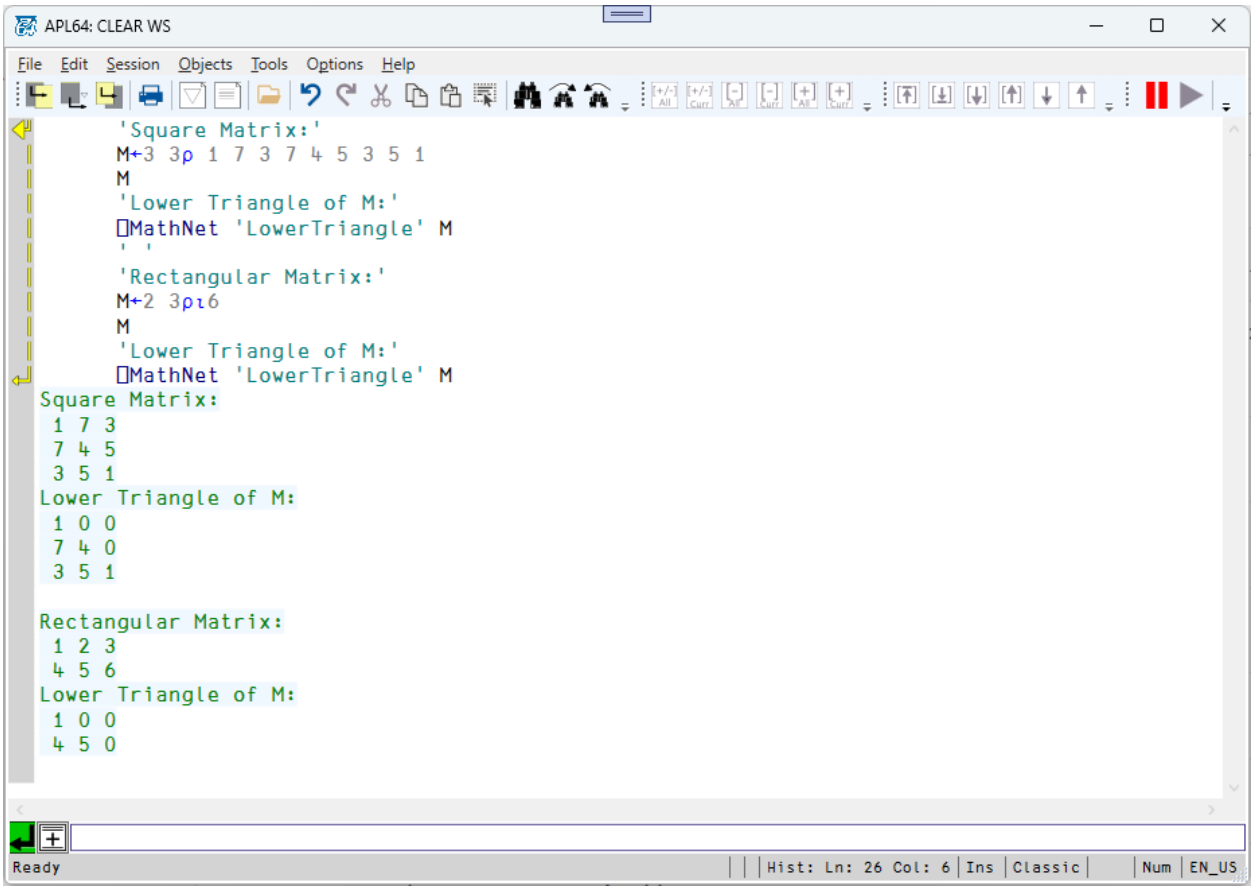

```
M←3 3⍴1 7 3 7 4 5 3 5 1
⎕MathNet 'L1Norm' M
```

L2Norm




```
M←3 3⍴1 7 3 7 4 5 3 5 1
⎕MathNet 'L2Norm' M
```

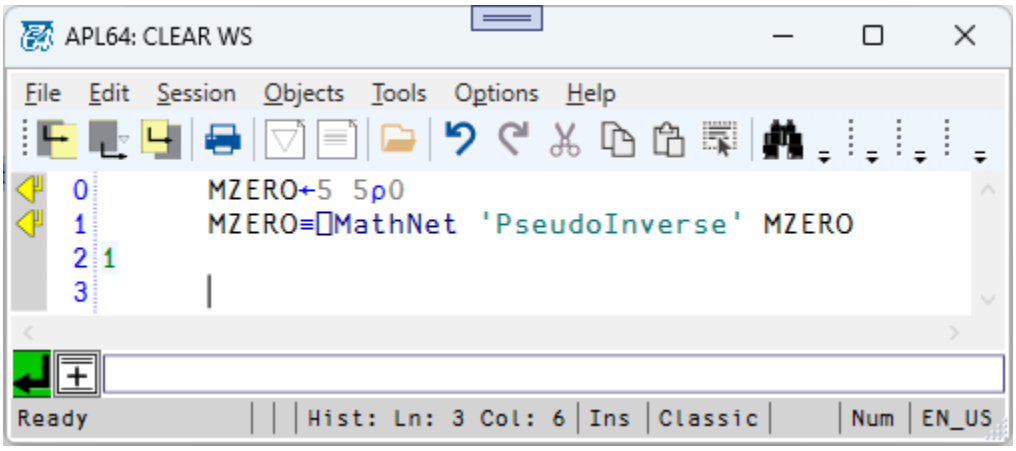
Lower Triangle

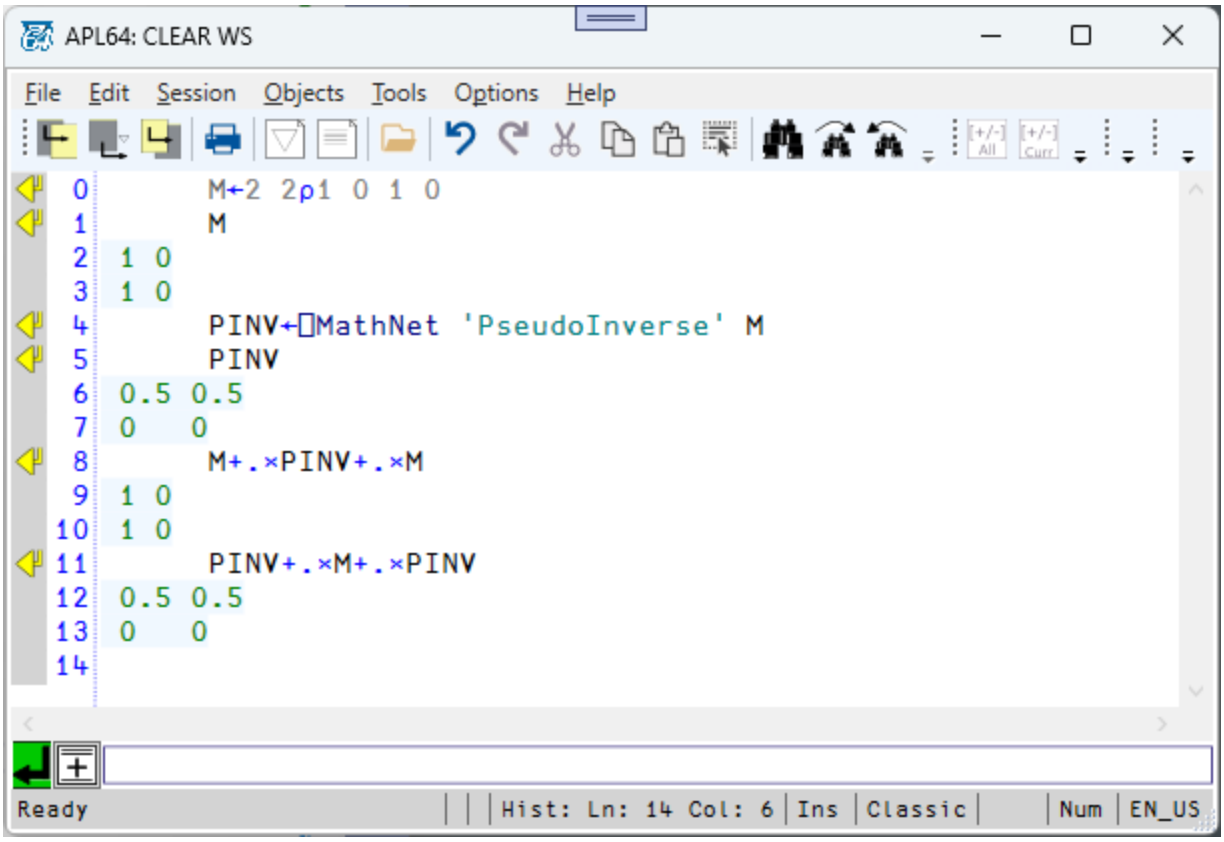


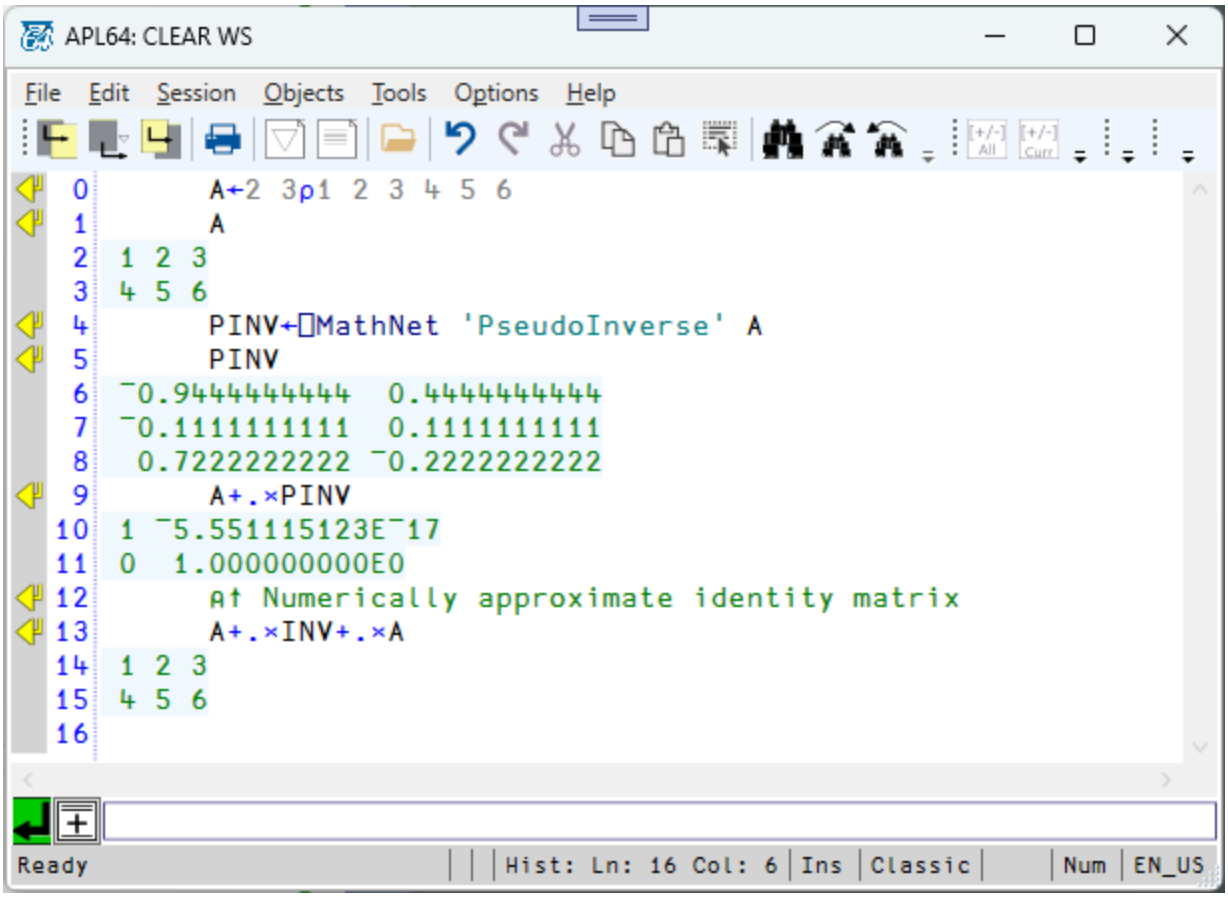

```
M←3 3⍴1 7 3 7 4 5 3 5 1
M
'LowerTriangle of M: '
⎕MathNet 'LowerTriangle' M
```

Pseudo Inverse

Any rectangular or square matrix has a [Pseudo Inverse](#).



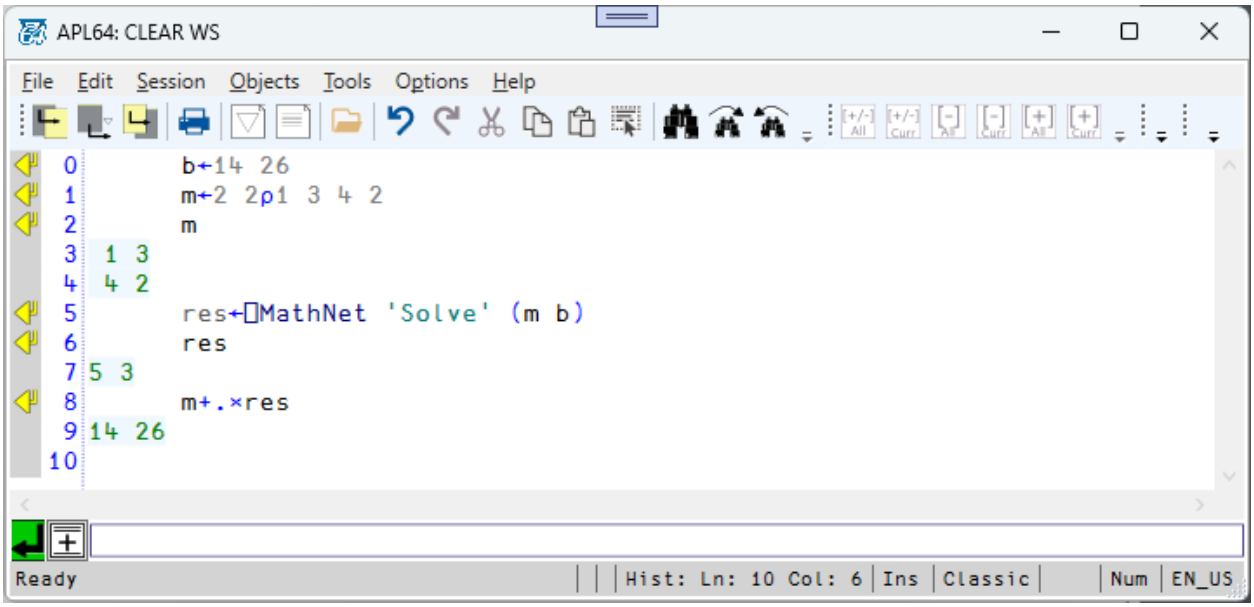




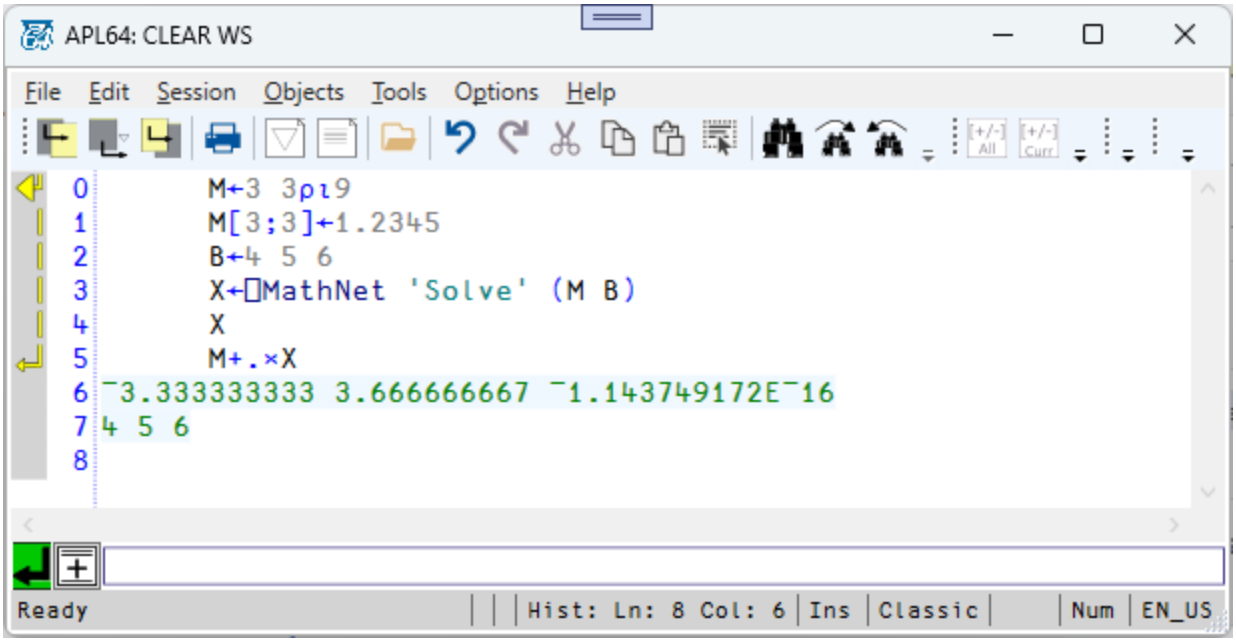
Solve

The Solve action applies to square, non-singular (invertible) matrices.

Find x and y satisfying the system of simultaneous linear equations: 1x + 3y = 14 and 4x + 2y = 26



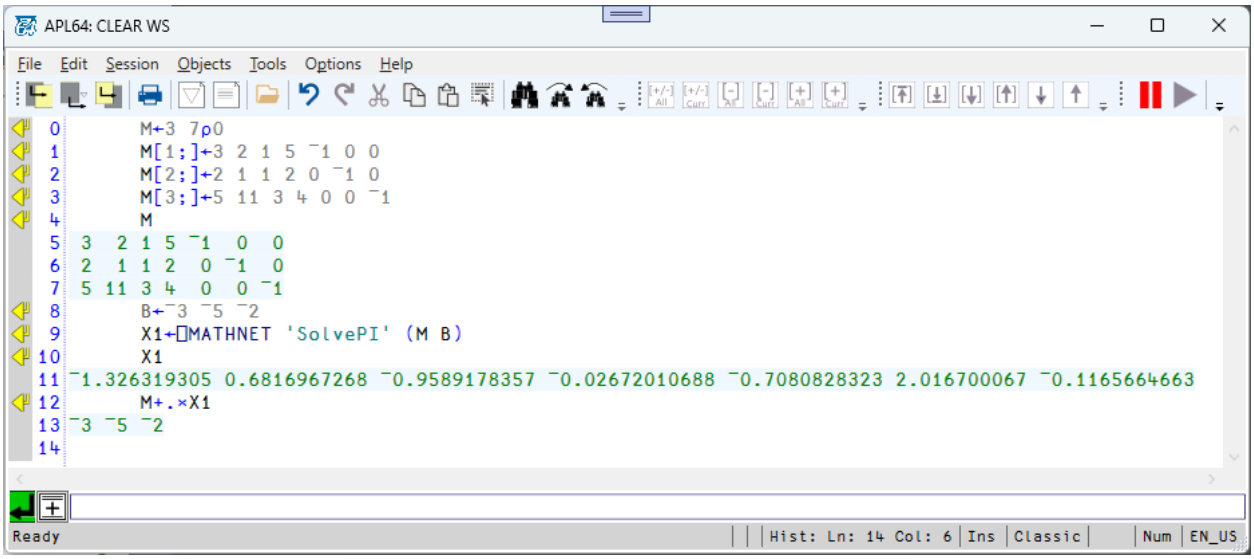
Solve for X where B = M+.×X



Solve PI

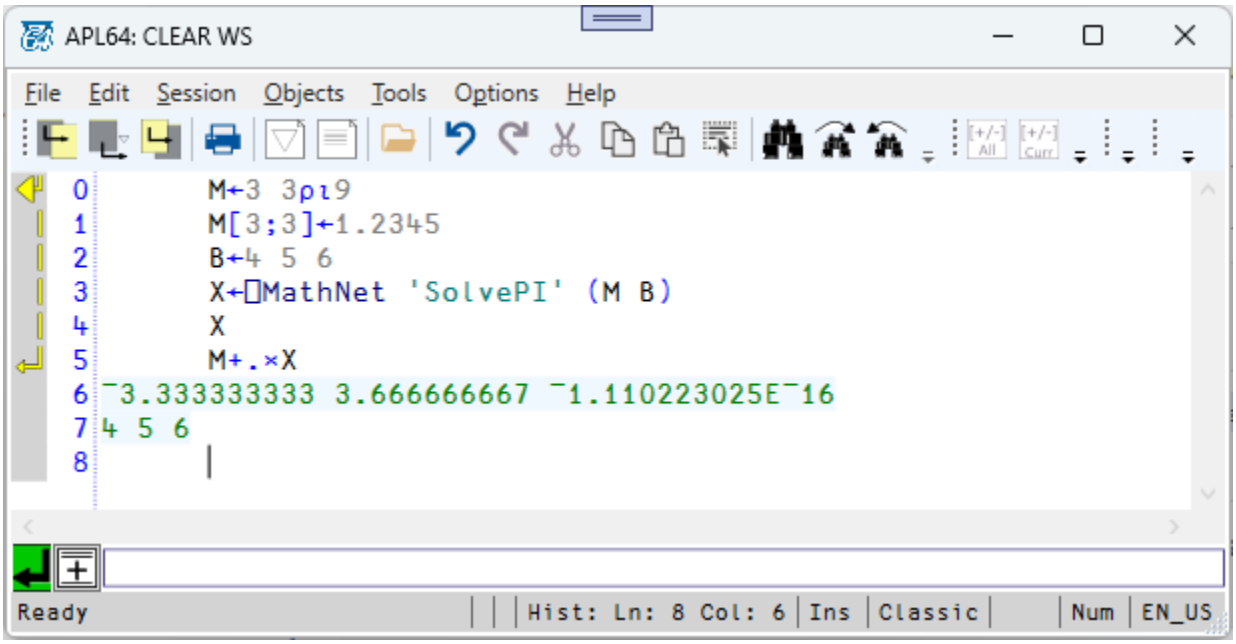
The SolvePI action applies to square or rectangular matrices and uses the pseudo inverse in its algorithm.

Solve for X1, where B = M+.×X1




```
M←3 7⍴0
M[1;]←3 2 1 5 ¯1 0 0
M[2;]←2 1 1 2 0 ¯1 0
M[3;]←5 11 3 4 0 0 ¯1
M
B←¯3 ¯5 ¯2
X1←⎕MATHNET 'SolvePI' (M B)
X1
M+.×X1
```

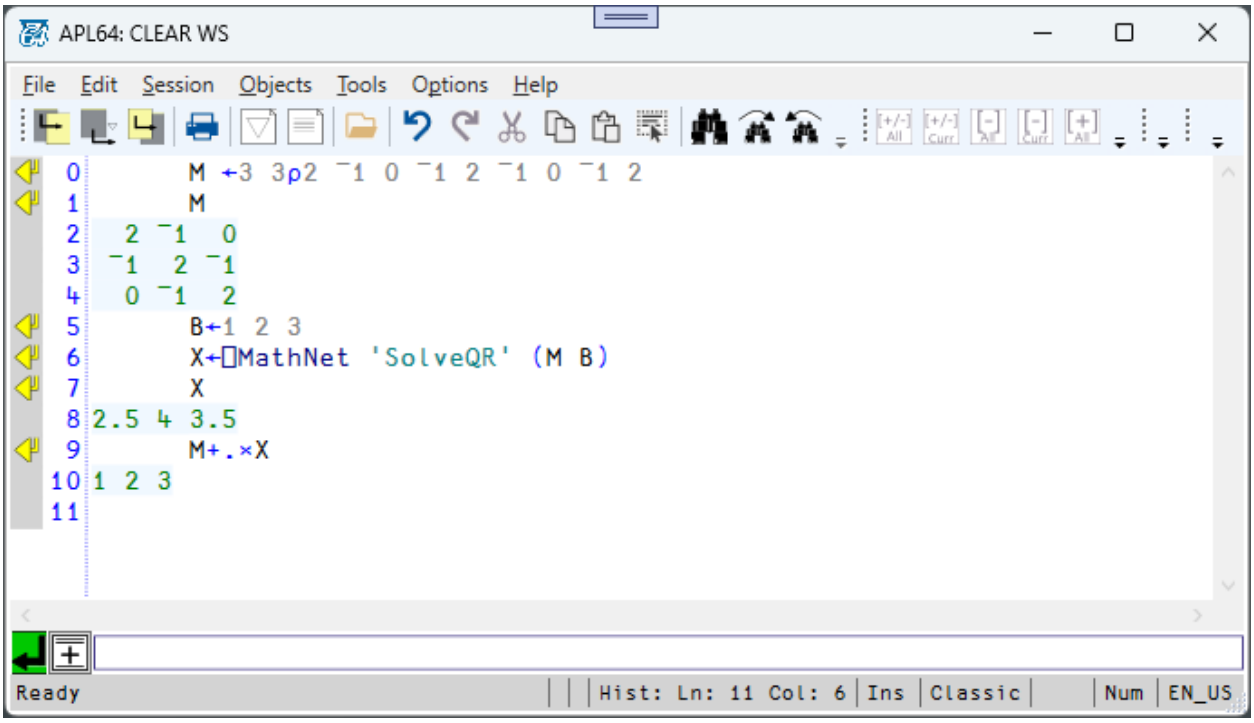
Solve for X where B = M+.×X



Solve QR

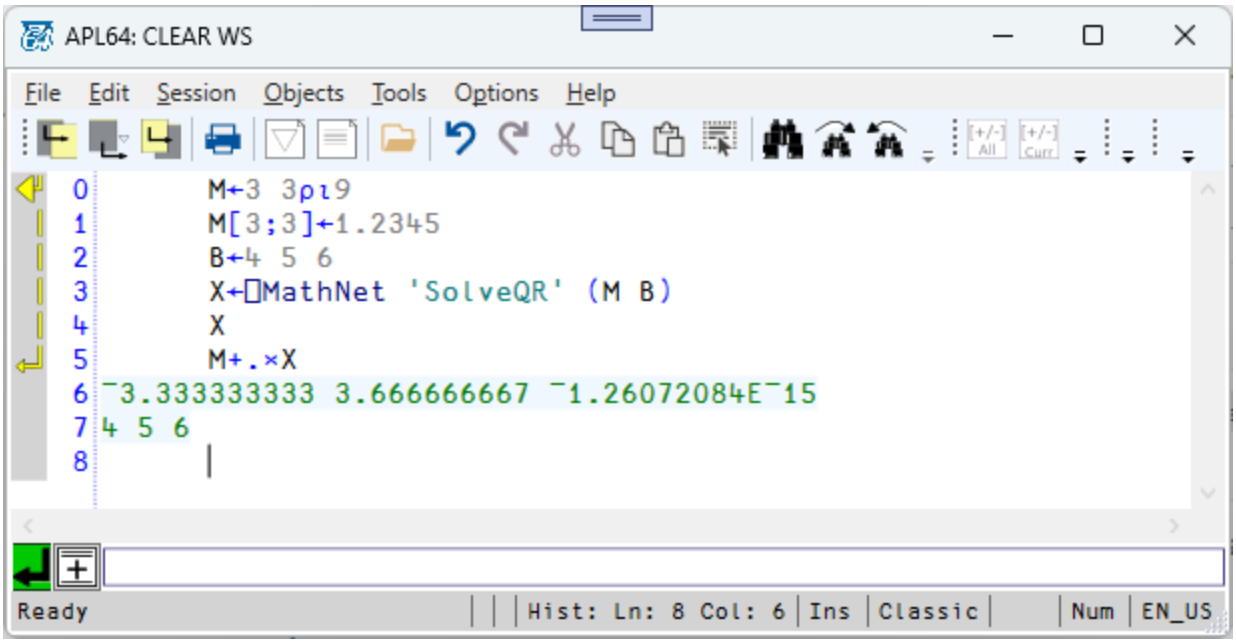
The SolveQR action applies to square matrices and uses the QR factorization in its algorithm.

Solve for X where B = M+.×X

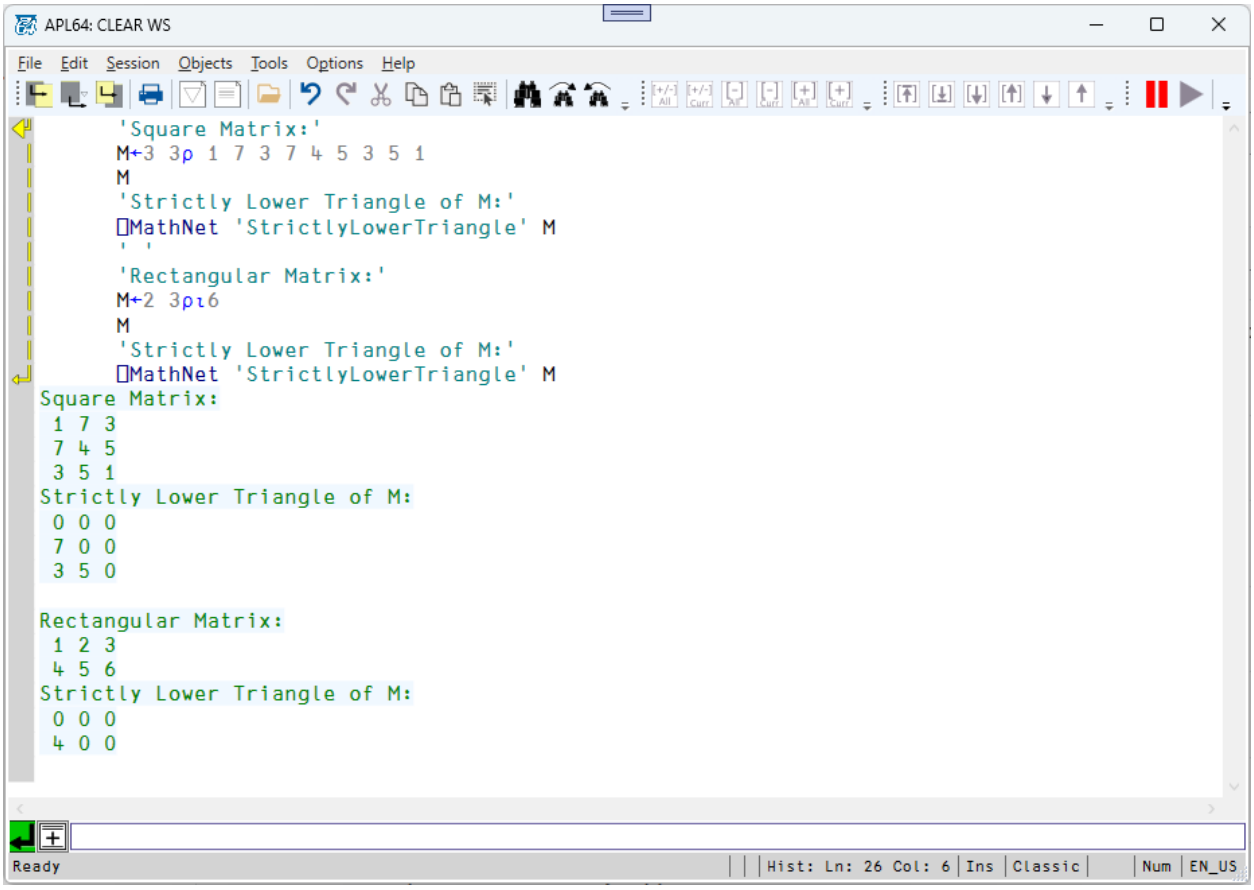



```
M←3 3⍴2 ¯1 0 ¯1 2 ¯1 0 ¯1 2
M
B←1 2 3
X←⎕MathNet 'SolveQR' (M B)
X
M+.×X
```

Solve for X where B = M+.×X

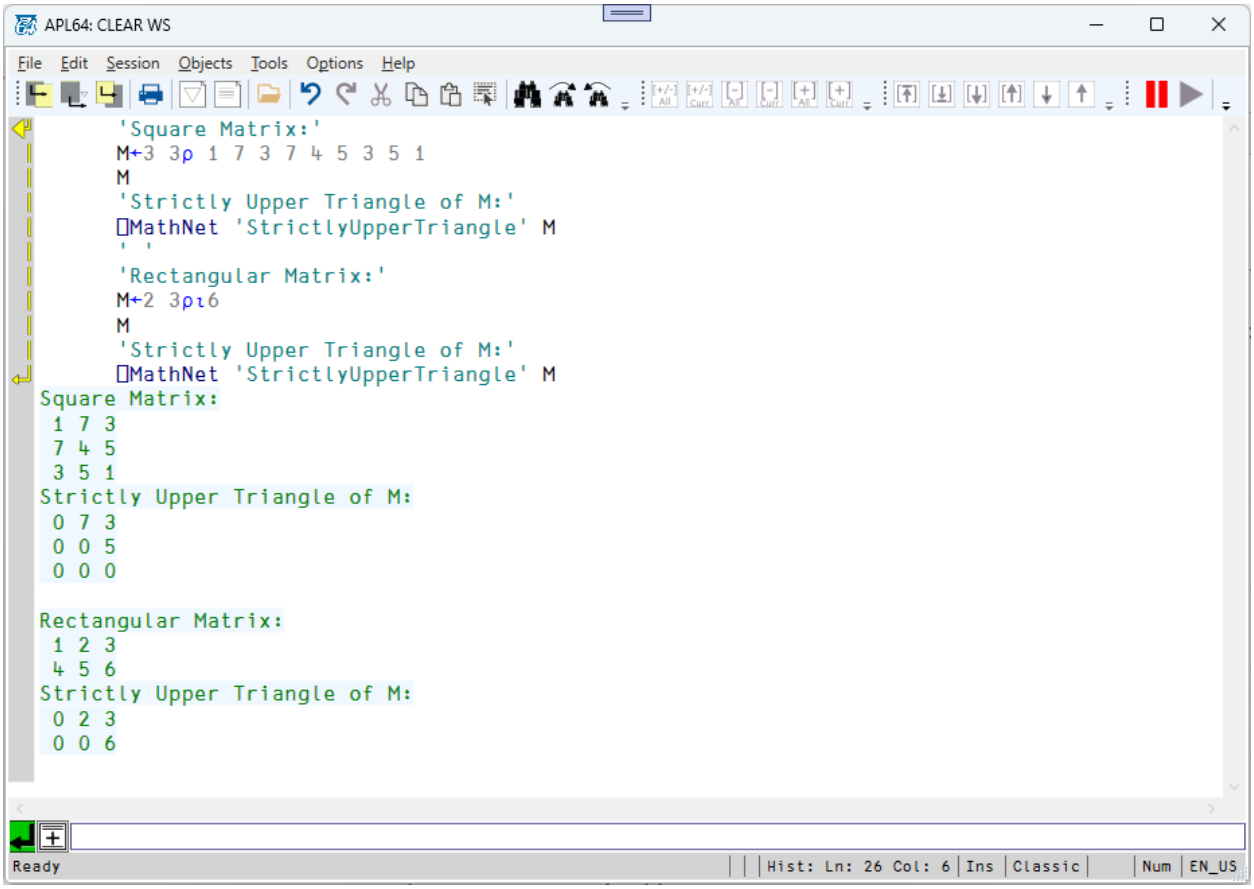


Strictly Lower Triangle



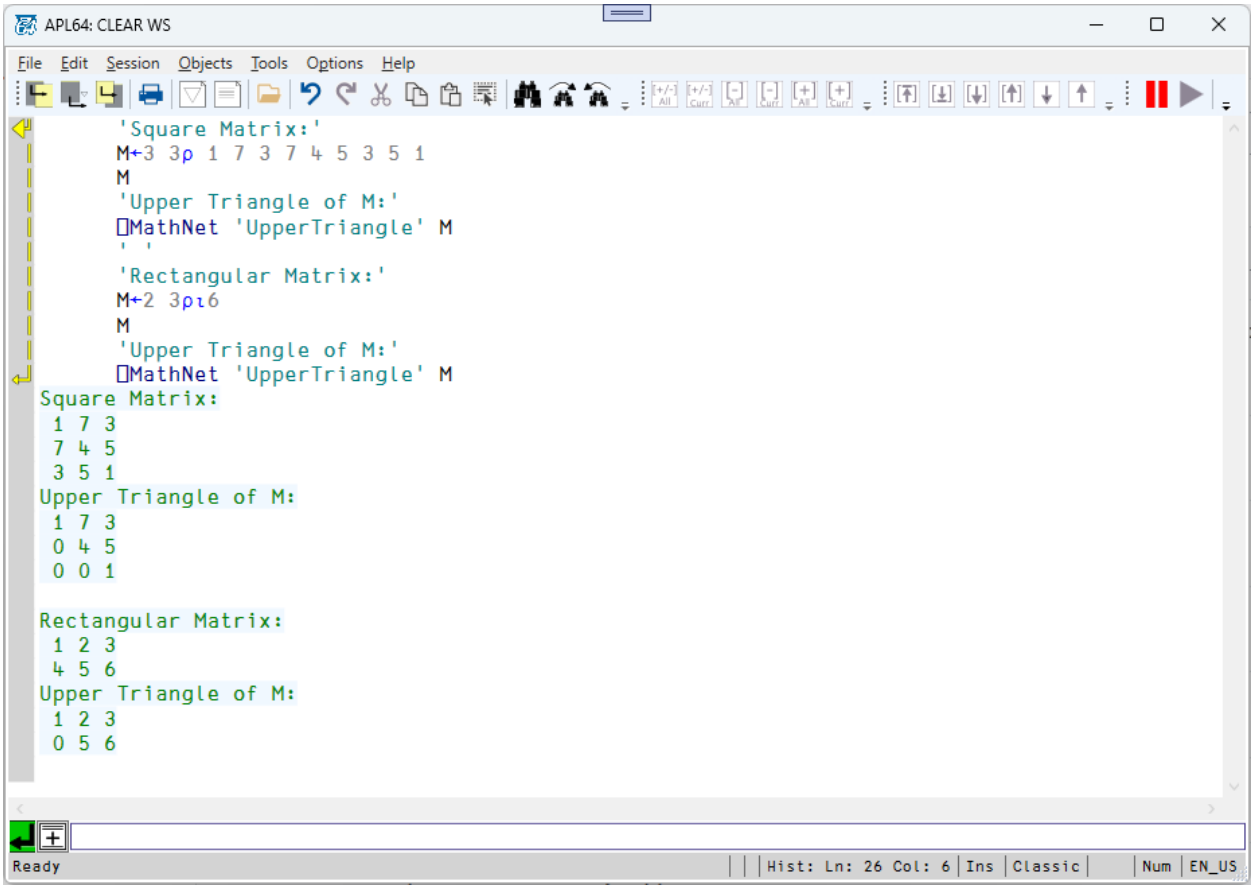

```
M←3 3⍴1 7 3 7 4 5 3 5 1
```

Strictly Upper Triangle




```
M←3 3⍴1 7 3 7 4 5 3 5 1
```

Upper Triangle




```
M←3 3⍴1 7 3 7 4 5 3 5 1
```