

Some Maple, MuPad and Matlab Examples

Creating a Matrix

The matrix $A = \begin{pmatrix} 2 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ will be used as an example.

`A:=Matrix([[2,2,3],[4,5,6],[7,8,9]]);` Create matrix A in Maple.
`A:=matrix([[2,2,3],[4,5,6],[7,8,9]]);` Create matrix A in MuPad.
`A=[2 2 3; 4 5 6; 7 8 9]` Create matrix A in Matlab.

Variable assignment uses `:=` in Maple and MuPad, whereas `=` is used in Matlab. The assignment creates a new symbol which is shorthand for the assignment, useful when you plan on using the matrix again. Uses are illustrated below.

The examples assume that two 3×3 matrices are stored into variables A and B .

Reduced Row Echelon Form

Maple: `linalg[rref](A);`
MuPad: `linalg::gaussJordan(A);`
Matlab: `rref(A)`

Multiply two matrices A and B

Maple: `A.B`
MuPad: `A*B`
Matlab: `A*B`

Inverse Matrix

Maple: `A^(-1);` or `1/A;`
Maple: `A^(-1);` or `1/A;`
MuPad: `A^(-1)` or `inv(A)`

Augment Matrices A and B

Maple: `<A|B>`
MuPad: `A.B`
Matlab: `[A B]`

Evaluate a Determinant

Maple: `linalg[det](A);`
MuPad: `linalg::det(A);`
Matlab: `det(A)`

Create an $n \times n$ identity matrix

Maple: `Matrix(n,n,shape=identity);`
MuPad: `matrix::identity(n);`
Matlab: `eye(n)`

The Maple command `with(linalg):` issued just once in your Maple document allows `linalg[rref](A)` to be shortened to `rref(A);`. Maple package `LinearAlgebra` is a richer package replacement for `linalg`.

Maple Package Equivalents

`linalg[det]` and `LinearAlgebra[Determinant]`

`linalg[rref]` and `LinearAlgebra[ReducedRowEchelonForm]`