Maple: Working with Matrices

Maple supports work and operations with matrices, on numerical and also symbolical level. Algorithms of special libraries as **NAG, LaPACK**, **ATLAS** and **MKL**, are utilised to provide accurate numerical calculations connected to matrix operations. Using a palette of commands for generation of matrices we can easily define any matrix of type e.g. 1000 by 1000 with entries in the form of decimal numbers. Basic command for matrix definition is the following:

```
> Matrix(r, c, init, ro, sc, sh, st, o, dt, f, a)
```

All parameters in the command are optional. Anyhow, system needs sufficient information, in order to create matrix structure. If there are no specifications included, answer of the system is matrix of type **0 x 0**.

Command

```
> Matrix(r)
```

generates matrix of type r x r, with entries defined implicitly as zeros.

```
> Matrix(2);
```

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Command in the form

```
> Matrix(r,c)
```

generates matrix of type r x c, with entries defined implicitly as zeros.

```
>  Matrix(2,3);
```

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Several further examples for matrix definition are provided:

```
> Matrix(1..3,1..2,5);
```

$$\begin{bmatrix} 5 & 5 \\ 5 & 5 \\ 5 & 5 \end{bmatrix}$$

```
> Matrix([[1,2,3],[4,5,6]]);
```

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

```
> f:= (i,j) -> x^(i+j-1):
> Matrix(2,f);
```

$$\begin{bmatrix} x & x^2 \\ x^2 & x^3 \end{bmatrix}$$

Entries of matrix in the previous example are defined as power functions of variable *x*, while its exponent is determined by the position of the entry in the matrix, ordered pair of numbers (*i*, *j*), for *i*, *j* = 1,2.

Matrix can be also defined by means of assignment operator. Matrix can be attached an arbitrary variable name and later it can be referred as this variable.

```
> A:=Matrix(3,4,[[1,2,3],[4,5,6]],readonly=true);
```

$$A := \begin{bmatrix} 1 & 2 & 3 & 0 \\ 4 & 5 & 6 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

```
> with(LinearAlgebra):
V := <<1,2,3>|<4,5,6>|<7,8,9>|<10,11,12>>;
```

$$V := \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix}$$

```
> MA := Matrix([[9,9,9,9],[9,9,9,9],[9,9,9,9],[9,9,9,9]]);
```

$$MA := \begin{bmatrix} 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 \end{bmatrix}$$

Operations with matrices

Product of matrix and scalar – number

Utilising the above defined matrix A, we will calculate its scalar multiple by 2, or 3/2. Operation can be easily coded by multiplication sign, star - *

```
> B:=2*A;
```

$$B := \begin{bmatrix} 2 & 4 & 6 & 0 \\ 8 & 10 & 12 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

> C:=(3/2)*A;

$$C := \begin{bmatrix} \frac{3}{2} & 3 & \frac{9}{2} & 0 \\ 6 & \frac{15}{2} & 9 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Operation can be also performed with the formal variable, e.g. λ., c, but using a different command:

> with(LinearAlgebra):
F:=ScalarMultiply(A,lambda/c);

$$F := \begin{bmatrix} \frac{\lambda}{c} & \frac{2\lambda}{c} & \frac{3\lambda}{c} & 0 \\ \frac{4\lambda}{c} & \frac{5\lambda}{c} & \frac{6\lambda}{c} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Sum, product and difference of matrices

Let us define matrices with formal variables as entries, and calculate consequently their sum, product and difference.

> M1:=Matrix([[cos(k*Pi*v),sin(k*Pi*v),0,0],[-sin(k*Pi*v),cos(k*Pi*v),0,0],[0,0,1,0],[0,0,0,1]]);

> M2:=Matrix([[1,0,0,0],[0,cos(l*Pi*v),sin(l*Pi*v),0],[0,-sin(l*Pi*v),cos(l*Pi*v),0],[0,0,0,1]]);

$$M1 := \begin{bmatrix} \cos(k\pi v) & \sin(k\pi v) & 0 & 0 \\ -\sin(k\pi v) & \cos(k\pi v) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M2 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(l\pi v) & \sin(l\pi v) & 0 \\ 0 & -\sin(l\pi v) & \cos(l\pi v) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

> M:=M1+M2;

$$M := \begin{bmatrix} \cos(k\pi v)+1 & \sin(k\pi v) & 0 & 0 \\ -\sin(k\pi v) & \cos(k\pi v)+\cos(l\pi v) & \sin(l\pi v) & 0 \\ 0 & -\sin(l\pi v) & 1+\cos(l\pi v) & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

```
> N:=M1.M2;
```

$$N := \begin{bmatrix} \cos(k\,\pi\,v) & \sin(k\,\pi\,v)\cos(l\,\pi\,v) & \sin(k\,\pi\,v)\sin(l\,\pi\,v) & 0 \\ -\sin(k\,\pi\,v) & \cos(k\,\pi\,v)\cos(l\,\pi\,v) & \cos(k\,\pi\,v)\sin(l\,\pi\,v) & 0 \\ 0 & -\sin(l\,\pi\,v) & \cos(l\,\pi\,v) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
> K:=M1-M2;
```

$$K := \begin{bmatrix} \cos(k\,\pi\,v) - 1 & \sin(k\,\pi\,v) & 0 & 0 \\ -\sin(k\,\pi\,v) & \cos(k\,\pi\,v) - \cos(l\,\pi\,v) & -\sin(l\,\pi\,v) & 0 \\ 0 & \sin(l\,\pi\,v) & 1 - \cos(l\,\pi\,v) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Operations with matrices, vectors and scalars can be also performed by means of LinearAlgebra package.

```
> with(LinearAlgebra):
Ax := <1.00004,1.99987,-0.00012>:
b := <1.,2.,0.>:
Add(Ax,b,1,-1);
```

$$\begin{bmatrix} 0.0000400000000000400036 \\ -0.0001299999999999963480 \\ -0.0001200000000000000004 \end{bmatrix}$$

Next command performing matrix product is Multiply

```
> s := <3|-2|7>;
```

$$s := [3, -2, 7]$$

```
> b := <x,y,z>;
```

$$b := \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

```
> d:= Multiply(s,b);
```

$$d := 3\,x - 2\,y + 7\,z$$

Calculation of inverse matrix

Inverse matrix can be formally denoted as matrix in power -1. Correctness of the inverse matrix calculation can be proved by calculating the product of the original matrix and its inverse matrix, with the result in the unit matrix E.

```
> K:=Matrix([[1,1,0,0],[0,0,1,1],[1,1,1,0],[0,1,0,1]]);
```

$$K := \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

```
> L:=K^(-1);
```

$$L := \begin{bmatrix} 2 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & 0 & 1 & 0 \\ 1 & 1 & -1 & 0 \end{bmatrix}$$

```
> E:=K.L;
```

$$E := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Calculation of the inverse matrix can be performed also by means of the command Inverse in mod *n*, or by means of the LinearAlgebra package command - MatrixInverse

```
> X := Matrix([[1,2,3],[1,3,0],[1,4,3]]);
```

$$X := \begin{bmatrix} 1 & 2 & 3 \\ 1 & 3 & 0 \\ 1 & 4 & 3 \end{bmatrix}$$

```
> Y := Inverse(X) mod 5;
```

$$Y := \begin{bmatrix} 4 & 1 & 1 \\ 2 & 0 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

```
> Z := X.Y;
```

$$Z := \begin{bmatrix} 11 & 10 & 10 \\ 10 & 1 & 10 \\ 15 & 10 & 16 \end{bmatrix}$$

```
> with(LinearAlgebra):
MatrixInverse(<<a,b>|<c,d>>);
```

$$\begin{bmatrix} \frac{d}{a\,d - c\,b} & -\frac{c}{a\,d - c\,b} \\ -\frac{b}{a\,d - c\,b} & \frac{a}{a\,d - c\,b} \end{bmatrix}$$

In some situations the result must be simplified.

```
> with(LinearAlgebra):
R := Matrix([[cos(alpha),-sin(alpha)],[sin(alpha),cos(alpha)]]);
```

$$R := \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

```
> MatrixMatrixMultiply(R, Transpose(R));
```

$$\begin{bmatrix} \cos(\alpha)^2 + \sin(\alpha)^2 & 0 \\ 0 & \cos(\alpha)^2 + \sin(\alpha)^2 \end{bmatrix}$$

```
> Map(simplify,%);
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Transposed matrix

The Transpose function returns the transpose of the input matrix. To illustrate calculation of a transpose matrix we will use the above defined matrix V.

```
> Transpose(V);
```

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix}$$

The same steps should be taken when using package LinearAlgebra. After definition of matrix M, with entries generated randomly, the transpose matrix can be calculated using functions Create, Transpose

```
> with(LinearAlgebra:-Modular):
M := Create(30,3,3,random,integer);
```

$$M := \begin{bmatrix} 27 & 29 & 6 \\ 18 & 6 & 19 \\ 3 & 1 & 24 \end{bmatrix}$$

```
> Transpose(30,M,inplace): M;
```

$$\begin{bmatrix} 27 & 18 & 3 \\ 29 & 6 & 1 \\ 6 & 19 & 24 \end{bmatrix}$$

Calculating determinant of a square matrix

Determinant of a matrix is number, which can be computed by using a Gauss row-reduction that functions correctly for any rank *m* of a square matrix. System Maple performs this long-lasting numerical calculation by a simple function Determinant. In the example below we firstly define lower triangular matrix M a then compute its determinant. In the last example, determinant of defined matrix N is computed, while matrix can be determined directly in the function for determinant computation.

```
> with(LinearAlgebra):
M := Matrix(3,[[a],[b,c],[d,e,f]],shape=triangular[lower]);
```

$$M := \begin{bmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{bmatrix}$$

```
> Determinant(M);
```

$$a\,c\,f$$

```
> N:=Matrix([[3,2,1,0],[7,6,5,4],[1,0,9,8],[5,4,3,2]]);
> Determinant(N);
```

$$N := \begin{bmatrix} 3 & 2 & 1 & 0 \\ 7 & 6 & 5 & 4 \\ 1 & 0 & 9 & 8 \\ 5 & 4 & 3 & 2 \end{bmatrix}$$

$$0$$

```
> Determinant(Matrix([[3,2,1,0],[7,6,5,4],[1,0,9,8],[5,4,3,2]]));
```

$$0$$