

Mathematical Modelling with Discrete Tools

„Ha egy probléma megoldásán fáradozol, nagy segítséget jelent, ha eleve ismered a megoldást.”
(Murphy)

The linear algebra is full of interesting mathematical tasks. Maple provides the LinearAlgebra package to execute the operations which belong to the field of linear algebra, such as vector and matrix operations, the calculation of determinants, the determination of eigenvalues and eigenvectors. We are going to review these by examining matrixes that have interesting properties. Of course, these matrixes are not magic just nice.

In order to deepen the knowledge of the problem-solving process, we are going to show the mathematical modelling with Maple by solving two tasks. Who would question that the shepherd, the wolf or the goat are not classical mathematical concepts? Not to mention the cabbage.

Magic matrixes

Determine the determinant, the inverse, the characteristic polynomial, the eigenvalues and the alters of each eigenvalue of the

$$\begin{bmatrix} M & 0 & 0 & 0 & M \\ 0 & A & 0 & 0 & 0 \\ 0 & 0 & P & 0 & P \\ 0 & 1 & 0 & L & 0 \\ 0 & 0 & 0 & 0 & E \end{bmatrix}$$

matrix.

The Matrix procedure, which is the number one tool to create matrixes, expects the row and column dimension of the matrix to be created as its first parameter. Its third parameter is the list of the elements of the matrix listed continuously

```
> Q := Matrix(5, 5, [[M, 0, 0, 0, M], [0, A, 0, 0, 0], [0, 0, P, 0, P], [0, 1, 0, L, 0], [0, 0, 0, 0, E]])
```

$$Q := \begin{bmatrix} M & 0 & 0 & 0 & M \\ 0 & A & 0 & 0 & 0 \\ 0 & 0 & P & 0 & P \\ 0 & 1 & 0 & L & 0 \\ 0 & 0 & 0 & 0 & E \end{bmatrix} \quad (1)$$

```
> whatype(Q)
```

Matrix (2)

The output of the whatype(Q) instruction shows that we are creating a Matrix-type object. Let's start the

exploring of the magic properties of this matrix by calculating its determinant. Call the Determinant procedure of the LinearAlgebra package with the Q parameter.

```

> with(LinearAlgebra):
> Determinant(Q)
MAPLE (3)

```

The value of the determinant is the product of the elements of the main diagonal. We have seen this previously. Let's continue with the determinants of the powers of the matrix.l.

```

> Q^2
MAPLE
[
  [M^2, 0, 0, 0, M^2 + ME],
  [0, A^2, 0, 0, 0],
  [0, 0, P^2, 0, P^2 + PE],
  [0, A + L, 0, L^2, 0],
  [0, 0, 0, 0, E^2]
]
(4)

```

```

> Determinant(Q^2)
M^2 A^2 P^2 L^2 E^2
(5)

```

```

> Determinant(Q^3)
M^3 A^3 P^3 L^3 E^3
(6)

```

That's it! Notice that the determinant of the nth power of the Q matrix is $(M.A.P.L.E)^n$.

Let's continue with the powers of the inverse matrix. The inverse of the Q can be created with the Q-1 expression if we do not want to use the MatrixInverse procedure of the LinearAlgebra package

```

> Q^-1
MAPLE
[
  [1/M, 0, 0, 0, -1/E],
  [0, 1/A, 0, 0, 0],
  [0, 0, 1/P, 0, -1/E],
  [0, -1/AL, 0, 1/L, 0],
  [0, 0, 0, 0, 1/E]
]
(7)

```

```

> Q^-2

```


The dot (.) infix operation is the operator of the non-commutative matrix multiplication.

Let's get back to the original task and keep on looking at the magic properties of the matrix. The determinant of the Q matrix is the product of the M,A,P,L,E variables and the inverse matrix also looks fine. Let's look at the characteristic polynomial of the Q. The CharateristicPolynomial procedure expects the matrix the characteristic polynomial of which it is creating as its first parameter. Its second parameter is the independent variable of the characteristic polynomial created. For this, we use the λ .

$$\begin{aligned}
 &> \text{CharacteristicPolynomial}(Q, \lambda) \\
 &\lambda^5 - (E + L + P + A + M) \lambda^4 - (-EL - PE - EA - ME - LP - AL - LM - PA - PM \\
 &\quad - AM) \lambda^3 - (ELP + EAL + ELM + EPA + EPM + EAM + LPA + LPM \\
 &\quad + LAM + PAM) \lambda^2 - (-ELPA - ELPM - ELAM - EPAM - LPAM) \lambda \\
 &\quad - MAPLE
 \end{aligned} \tag{11}$$

Other interesting relations have come up. The free term of the characteristic polynomial is the product of the M,A,P,L,E variables and the coefficient of the λ^i is created by generating all the possible subsets of the (5-i) element of the M,A,P,L,E set. Then multiply these elements and add the terms created. However, let's look at the roots, that is, the eigenvalues of the Q.

$$\begin{aligned}
 &> \text{factor}(\%) \\
 &\quad -(E - \lambda) (L - \lambda) (P - \lambda) (-\lambda + A) (M - \lambda)
 \end{aligned} \tag{12}$$

$$\begin{aligned}
 &> \text{Eigenvalues}(Q) \\
 &\quad \begin{bmatrix} P \\ M \\ L \\ E \\ A \end{bmatrix}
 \end{aligned} \tag{13}$$

The eigenvalues are M,A,P,L,E. It is not surprising though a little cumbersome that the Eigenvalues procedure does not list the values of the components of the vector received as the result.

$$\begin{aligned}
 &> \text{Eigenvectors}(Q, \text{output}='list') \\
 &\left[\left[\left[\begin{bmatrix} -\frac{M}{-E+M} \\ 0 \\ -\frac{P}{-E+P} \\ 0 \\ 1 \end{bmatrix} \right], \left[\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \right], \left[\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right], \left[\begin{bmatrix} 0 \\ A-L \\ 0 \\ 1 \\ 0 \end{bmatrix} \right], \left[\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right] \right], \left[\begin{bmatrix} E, 1, \end{bmatrix}, \left[\begin{bmatrix} L, 1, \end{bmatrix}, \left[\begin{bmatrix} M, 1, \end{bmatrix}, \left[\begin{bmatrix} A, 1, \end{bmatrix}, \left[\begin{bmatrix} P, 1, \end{bmatrix} \right] \right] \right] \right]
 \end{aligned} \tag{14}$$

$$\left[\begin{array}{c} \left[\begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{array} \right] \end{array} \right]$$

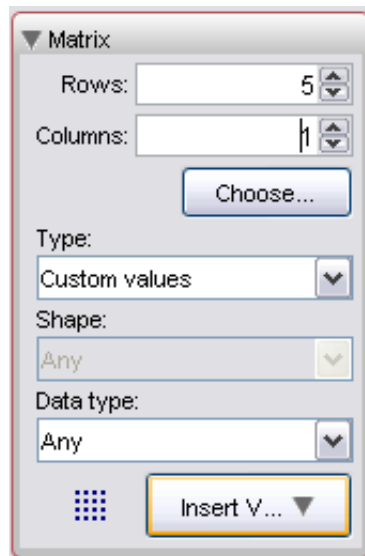
The Eigenvectors does not only calculate the eigenvalues but it also calculates their multiplicity and a base of a subspace belonging to them. The instruction results in a series of [s,m,b] syntax lists in which case the s is the eigenvalue, the m is its multiplicity and the b is the set of vectors which gives the base of the subspace belonging to the s eigenvalue. According to this, for example, the A eigenvalue is the single multiplicity root of the characteristic polynomial and the one dimensional subspace belonging to it is created by the [0,A-L,0,1,0] vector.

In one of his famous aporias, Zeno of Elea denied motion. Zeno was one of the most talented philosophers of the 5th century B.C. who was making a good living from thinking and expressing his opinions about the matters of the world. He realised Galilei's theory of relativity, that is, a spectator moving with v1 speed perceives the speed of an object approaching him with v2 speed as v1+v2. So is the speed of the object v2 or v1+v2? It is impossible that both cases exist at the same time, says Zeno, so the conclusion is that motion does not exist. And he was even given money for this at that time.

Now we are going to use the letters of Zeno to show some procedures of the LinearAlgebra package for other matrixes.

$$\left[\begin{array}{l} > b := \text{Vector}([Z, E, n, O, N]) \\ \\ b := \begin{bmatrix} Z \\ E \\ n \\ O \\ N \end{bmatrix} \end{array} \right. \quad (15)$$

The Vector (written with capital V) procedure creates a one-dimensional array the elements of which are the elements of the list given as the parameter. The Vector cannot only create column and row vectors but it also has several useful options which can make the work with the system really efficient. Vectors can also be created by using the Matrix palette.



By giving the values 5 and 1 to the Rows and Column fields, then clicking the InsertV button the system inserts a column vector of 5 components onto the worksheet in which case the components are placeholders which we can overwrite with the values we want (use TAB for stepping forth and SHIFT+TAB for back).

The Matrix palette also supports the creation of matrixes. By giving the numbers of the rows and the columns of the matrix to be created we specify the size of the matrix. This can be done another way as well, by taking the pointer above the Choose button and holding down the left button. As a result, a 10x10 square is displayed on the screen where the squares located in the upper left-side corner can be coloured to blue by moving the pointer. The coloured area is always coherent and when releasing the button of the mouse, the system writes the number of the blue rows and columns into the Rows and Columns fields. During moving, the pressing of SHIFT changes the fields into green and the CTRL changes them to red. In the case of the green fields, the number of the rows and columns are multiplied by 10, and in the case of the red fields, it is multiplied by 100.

We can give with what kind of values Maple should fill the matrix created in the Type field. These can be 0 (Zero filled), 1 (One filled), Identity or Random matrixes. We can set in the Shape field whether we want to create a diagonal or a triangular matrix. And we can set the data types to be applied in the Data Type field.

As an example, let's create the

$$\begin{bmatrix} M & A & P & L & E \\ -M & 0 & 0 & 0 & 0 \\ 0 & -A & 0 & 0 & 0 \\ 0 & 0 & -P & 0 & 0 \\ 0 & 0 & 0 & -L & 0 \end{bmatrix}$$

matrix. This is a 5x5 matrix with lots of zero elements. That's why after giving the measures, choose the Zero filled list element from the Type list. Change the elements in the matrix inserted. Notice that we

have not chosen the Custom values element from the Type list thus the elements of the matrix generated will not be placeholders that's why the TAB does not work here. The cursor has to be moved above the elements so that they could be changed.

If you had not been able to execute this task then do the following instruction.

$$\begin{array}{l}
 \text{> } Q := \text{Matrix}(5, 5, [[M, A, P, L, E], [-M, 0, 0, 0, 0], [0, -A, 0, 0, 0], [0, 0, -P, 0, 0], [0, 0, 0, -L, 0]]) \\
 \\
 Q := \begin{bmatrix} M & A & P & L & E \\ -M & 0 & 0 & 0 & 0 \\ 0 & -A & 0 & 0 & 0 \\ 0 & 0 & -P & 0 & 0 \\ 0 & 0 & 0 & -L & 0 \end{bmatrix}
 \end{array} \tag{16}$$

We have a 5x5 Q matrix and a b column vector consisting of 5 elements. What if we consider the Q as the matrix of a linear system of equation and the b as the column vector of the free terms located on the right side of the equality signs.

$$\begin{array}{rcl}
 M \cdot x_1 + A \cdot x_2 + P \cdot x_3 + L \cdot x_4 + E \cdot x_5 & = & Z \\
 -M \cdot x_1 & = & E \\
 \quad -A \cdot x_2 & = & n \\
 \quad \quad -P \cdot x_3 & = & O \\
 \quad \quad \quad -L \cdot x_4 & = & N
 \end{array}$$

Let's solve the linear system of equation. Its determinant is

$$\begin{array}{l}
 \text{> } \text{Determinant}(Q) \\
 \\
 \text{MAPLE}
 \end{array} \tag{17}$$

so Q is also a magic matrix. We can also assume that each of the M,A,P,L,E variables is not zero thus according to Cramer's rule the system of equation has an obvious solution. Let's consider T as the additional matrix of the system of equation.

$$\begin{array}{l}
 \text{> } T := \langle Q | b \rangle \\
 \\
 T := \begin{bmatrix} M & A & P & L & E & Z \\ -M & 0 & 0 & 0 & 0 & E \\ 0 & -A & 0 & 0 & 0 & n \\ 0 & 0 & -P & 0 & 0 & O \\ 0 & 0 & 0 & -L & 0 & N \end{bmatrix}
 \end{array} \tag{18}$$

Let's look for the solution of the system of equation with the simplified version of the Gaussian elimination. Disregard if one of the elements located in the main diagonal becomes zero during the process or not because we have tried it and the answer is no.

```

> U:=T: print(U);
for i from 1 to 4 do
  for j from i+1 to 5 do
    U:=RowOperation(U,[j,i],-U[j,i]/U[i,i])
  
```

```

od:
print(U);
od;

```

$$\begin{bmatrix} M & A & P & L & E & Z \\ -M & 0 & 0 & 0 & 0 & E \\ 0 & -A & 0 & 0 & 0 & n \\ 0 & 0 & -P & 0 & 0 & O \\ 0 & 0 & 0 & -L & 0 & N \end{bmatrix}$$

$$\begin{bmatrix} M & A & P & L & E & Z \\ 0 & A & P & L & E & Z+E \\ 0 & -A & 0 & 0 & 0 & n \\ 0 & 0 & -P & 0 & 0 & O \\ 0 & 0 & 0 & -L & 0 & N \end{bmatrix}$$

$$\begin{bmatrix} M & A & P & L & E & Z \\ 0 & A & P & L & E & Z+E \\ 0 & 0 & P & L & E & n+Z+E \\ 0 & 0 & -P & 0 & 0 & O \\ 0 & 0 & 0 & -L & 0 & N \end{bmatrix}$$

$$\begin{bmatrix} M & A & P & L & E & Z \\ 0 & A & P & L & E & Z+E \\ 0 & 0 & P & L & E & n+Z+E \\ 0 & 0 & 0 & L & E & O+n+Z+E \\ 0 & 0 & 0 & -L & 0 & N \end{bmatrix}$$

$$\begin{bmatrix} M & A & P & L & E & Z \\ 0 & A & P & L & E & Z+E \\ 0 & 0 & P & L & E & n+Z+E \\ 0 & 0 & 0 & L & E & O+n+Z+E \\ 0 & 0 & 0 & 0 & E & N+O+n+Z+E \end{bmatrix}$$

(19)

Let's start with the T matrix. During the first run of the outer cycle we zero the first column of the matrix in a way that we add the appropriate number fold of the first line to the second, third, fourth and fifth line. This task is executed by the inner cycle. After this, repeat the process but in this case concentrate on the Q2,2 element and our aim is to zero the column below it. This is executed by the second run of the outer cycle. Repeat these processes three times and finally you will get a fine and regular upper triangular matrix.

Notice the changes of the elements located in the 6th column during the Gaussian elimination. We do believe that this output is so expressive and mathematically and didactically useful that it does not

require further explanations.

We also have to get to know the RowOperation procedure as well. The RowOperation(U,[i,j]c) instruction adds the c fold of the jth line to the ith line of the matrix and the matrix created is going to be the output of the instruction. Notice that the RowOperation does not overwrite the U matrix. While it is creating the result of its operation, it leaves the U unchanged. Of course, you can get it to do the opposite but for this the inplace option is to be used. In this case, the syntax of the instruction is RowOperation(U, [i,j]c, inplace).

We can get the solutions from the triangle matrix created during the elimination by backward substitution. Of course, Maple offers a procedure for this, called the BackwardSubstitute.

```
> BackwardSubstitute(U)
```

$$\begin{bmatrix} -\frac{E}{M} \\ -\frac{n}{A} \\ -\frac{O}{P} \\ -\frac{N}{L} \\ \frac{N + O + n + Z + E}{E} \end{bmatrix} \quad (20)$$

As a result, we get the solution of the initial system of equation in the form of a five-element list. The solutions show magic regularities. By reading the denominators, you can see the word Maple and the numerator of the last component is also interesting where we can find the sum of the letters of the word ZEnON. By the way, we could have received this result with the LinearSolve(Q,b) instruction in one step.

Zeno is said to have got on his peers' nerves. It is not surprising given the facts we know about him. Once, when they were discussing the existence of motion, one of his infuriated peers started to walk up and down the room like a trapped animal in a cage. Zeno was able to forbear this for a while then he snarled at his peer.

- Would you stop for a minute?

- So you have recognized that I'm in motion – snapped the infuriated peer at him. That's all about the deepness of the philosophical exchanges.

Now we are going to reveal the magic matrixes. Their creation is based on a simple idea. We know from linear algebra that if we multiply one row (or column) of the determinant with a number then its value changes to its number fold. So if we multiply the first row of the 5x5 determinant with M, its second row with A, and its third, fourth and fifth rows with P, L and E then the value of the determinant received is going to be the M.A.P.L.E fold of the value of the original determinant. So we only need a matrix the determinant of which is 1. It is even better if the matrix consists of 0 and 1 elements.

Where can we find such a matrix? Well, one way is to grab a collection of linear algebra tasks and look for the appropriate matrixes among the solutions. Btu we can also use the RandomMatrix procedure of Maple. Its first two parameters are the row and column dimensions of the matrix. Its third parameter is the entries=rand(a..b) option which makes the elements of the matrix created be arbitrary numbers within the [a,b] interval. So don't be surprised if you get another result after having the same run

```

> R := RandomMatrix(5, 5, generator = 0..1)

```

$$R := \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (21)$$

Now we have to multiply the rows of the R by the M,A,P,L,E variables. The following cycle executes this. In this case, the RowOperation(R,i,Yi) command multiplies the ith line of the R matrix by Yi while leaving the R unchanged. That's why we load the result onto the R matrix. Finally, calculate the determinant of the matrix.

```

> Y:=[M,A,P,L,E]:
  for i to 5 do R:=RowOperation(R,i,Y[i]) od:
> R

```

$$R = \begin{bmatrix} 0 & 0 & 0 & M & 0 \\ 0 & 0 & A & A & A \\ 0 & 0 & P & P & 0 \\ L & L & L & 0 & 0 \\ 0 & E & E & 0 & 0 \end{bmatrix} \quad (22)$$

```

> Determinant(R)

```

$$MAPLE \quad (23)$$

We have the philosopher's stone in our hands but do not take it too seriously. What you should take seriously is the knowledge acquired during the task.

What Have You Learnt About Maple?

- We have become familiar with several matrix operations during the worksheet. Some of these operations are top level procedures, that is, they can be immediately reached after the starting of the system while the others can be found in the LinearAlgebra package so if we do not want to refer to them with their long names then we have to give the with(LinearAlgebra) command.
- The Matrix procedure created a Matrix data type object. The procedure has 12 parameters all of which are optional to use. Thus we cannot show the whole scale here. Instead, we are going to show a simple but frequently used call sequence. In the
 - $\text{Matrix}(m,n, \text{the list of the lists})$
- instruction the m and the n are the numbers of the rows and columns of the matrix to be created and the list of the lists contains the initial values of the matrix. The list of the lists can contain an m list at best and every list can be an n element at best. These lists determine the start values of the rows of the matrix. If a list contains less than n elements then Maple fills the other elements with zeros. So this command creates an mxn matrix.

- The creation of the matrixes in a column continuous way can be executed by the $\langle\langle 1_oszlop_elemei \rangle, \langle 2_oszlop_elemei \rangle, \dots \rangle$ instruction. For example, $\langle\langle 1, 1, 1 \rangle, \langle 2, 2, 2 \rangle \rangle$ creates a matrix consisting of 3 rows and 2 columns.
- The number of the rows and the columns of the matrix can be determined by the `RowDimension` and the `ColumnDimension` procedures. Both of them can be found in the `LinearAlgebra` package.
- The dot (`.`) infix operator is the sign of the non-commutative product of the matrixes.
- The inverse of the M matrix can be calculated with the $\frac{1}{M}$ or the `M-1` expressions.
- The identity matrix can be created with the `IdentityMatrix` procedure. Its easiest call is the `IdentityMatrix(n)` which generates an $n \times n$ identity matrix.
- The characteristic polynomial of the M matrix can be created with the `CharacteristicPolynomial(M, variable)` instruction.
- The characteristic polynomial, the eigenvalues and the eigenvectors of the M matrix can be calculated with the `CharacteristicPolynomial`, the `EigenValues` and the `EigenVectors` procedures. All of these procedures can be found in the `LinearAlgebra` package.
- The multiplication of the matrixes and vectors by the scalar can be executed by the common multiplication operation (`*`).
- The `Vector` procedure creates a one-dimensional array the elements of which can be indexed with the integers of the $1..n$ domain. In this case the vector is n dimensional.
- The instruction with the `<A|B>` syntax executes the horizontal concatenation of the matrixes and vectors. The instruction assumes that the A and B matrixes consist of the same amount of rows ($RowDimension(A) = RowDimension(B)$). The first row of the result matrix is created in a way that the system writes the first row of B after the first row of the A . And it does the same for all the other rows. If B is a vector then the elements of B are copied subsequent to the rows of the A matrix.

• With the different calls of the `RowOperation` procedure, we can execute the elementary changes on the rows of the matrixes.

The `RowOperation(M, [i, j])` instruction exchanges the i th and j th rows of the M .

The `RowOperation(M, i, c)` instruction multiplies all the elements of the i th row of the M by the c .

The `[RowOperation(M, [i, j], c)` instruction adds the c fold of the j th row to the i th row.

The instructions leave the M matrix unchanged unless we use the `inplace` option.

- A **ColumnOperation** eljárás különböző hívásaival valósíthatjuk meg a matrixok oszlopain végezhető elemi átalakításokat.

A `ColumnOperation(M, [i, j])` utasítás felcseréli az M i -dik és j -dik oszlopát.

A `ColumnOperation(M, i, c)` utasítás az M i -dik oszlopának minden elemét c -vel szorozza.

A `ColumnOperation(M, [i, j], c)` utasítás a j -dik oszlop c -szeresét hozzáadja az i -dik oszlophoz.

Az utasítások magát az M mátrixot változtatlanul hagyják, kivéve, ha használjuk az **inplace** opciót.

- With the different calls of the `ColumnOperation` procedure, we can execute the elementary changes on

the columns of the matrixes.

The $ColumnOperation(M, [i, j])$ instruction exchanges the i th and j th columns of the M .

The $ColumnOperation(M, i, c)$ instruction multiplies all the elements of the i th column of the M by the c .

The $ColumnOperation(M, [i, j], c)$ instruction adds the c fold of the j th column to the i th column.

The instructions leave the M matrix unchanged unless we use the inplace option.

- The LinearSolve procedure is used to solve the system of equation with the $Ax=b$ syntax.
- The RandomMatrix procedure is used to generate arbitrary matrixes. The $RandomMatrix(n,m)$ is an $n \times m$ arbitrary matrix with the elements between -99 and 99 . The domain of the arbitrary numbers can be controlled with the generator option. The $RandomMatrix(n,m, generator=a..b)$ instruction generates elements within the $a..b$ domain and in case both a and b are integers, it generates integers. However, if one of them is a floating-point number, then floating-point values are created.

Exercise

1. Examine the magic properties of the following R_1 and R_2 matrixes

$$R_1 = \begin{bmatrix} M & A & P & L & E \\ M & 2.A & P & L & E \\ M & A & 2.P & L & E \\ M & A & P & 2.L & E \\ M & A & P & L & 2.E \end{bmatrix} \quad R_2 = \begin{bmatrix} M & -1 & 0 & 0 & 0 \\ A & 1 & -1 & 0 & 0 \\ P & 0 & 1 & -1 & 0 \\ L & 0 & 0 & 1 & -1 \\ E & 0 & 0 & 0 & 1 \end{bmatrix}$$

2. Prove that for the

$$M = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \\ x_1 & k.x_2 & x_3 & x_4 & x_5 \\ x_1 & x_2 & k.x_3 & x_4 & x_5 \\ x_1 & x_2 & x_3 & k.x_4 & x_5 \\ x_1 & x_2 & x_3 & x_4 & k.x_5 \end{bmatrix}$$

matrix it is $det(M) = k^4 x_1 x_2 x_3 x_4 x_5$ by elementary changes, by using the RowOperation and ColumOperation procedures subsequently. (Hint: enter the matrix then extract the first row from the second, the third, the fourth and the fifth rows.

3. Calculate the matrix of the R_1 and the inverse of the matrix of the previous task
4. Calculate the inverse of the R_2 matrix and denote it with R . Prove that the R matrix shows regularities if

a) $R_{i,i} - R_{i,i-1} = 1$ for every i, \square

b) the elements located on the right side of the main diagonal coincide with the element

located in the main diagonal in every row of the R,

c) the elements located on the left side of the main diagonal coincide with each other in every row of the R.

5. Solve the following equation with the method shown in the worksheet.

$$Mx_1 - x_2 = M;$$

$$Ax_1 + x_2 - x_3 = A;$$

$$Px_1 + x_3 - x_4 = P;$$

$$Lx_1 + x_4 - x_5 = L;$$

$$Ex_1 + x_5 = E;$$

6. By using the RowOperation and ColumnOperation procedures correct the elimination procedure of the worksheet. During the elimination examine if the element in the following main diagonal is zero and if yes search for a non-zero element in the column below it then exchange the two rows determined in this way.