Linear Algebra

- This is how we introduce vectors and matrices:

  `vector([1,5,0])` the output is a row vector.

  `vector([1,5,0]).transpose()` the output is a column vector.

  When multiplying `vector([1,5,0])` with a matrix, Sage knows if it is necessary to take it as a row or column vector, so it is not necessary to use transpose.

  `matrix([[1,5,0],[-1,-5,0]])` this is the 2x3 matrix whose first row is [1,5,0] and second row is [-1,-5,0].

  `matrix([[1,5,0],[-1,-5,0]]).transpose()` this is the 3x2 matrix whose first column is [1,5,0] and second column is [-1,-5,0].

  `diagonal_matrix([1,2,3])` this is the 3x3 diagonal matrix with the elements 1,2,3 on the main diagonal

  `identity_matrix(3)` this is the 3x3 identity matrix

  Once a matrix A is introduced, its elements are `A[0,0], A[0,1]` …

- This is how we compute the determinant, the inverse, the eigenvalues, the eigenvectors, the characteristic polynomial of a square matrix A:

  `A.det()`

  `A^(-1)`

  `A.eigenvalues()` the result is a vector that contains the eigenvalues of A

  `A.eigenvectors_right()` the result gives a list containing eigenvalues of A, the corresponding linearly independent eigenvectors of A , and the multiplicity.

  `A.charpoly('r')` we can specify the name of the variable, here is r

- Multiplication of two matrices A and B is: `A*B`

- This is how we compute the Jordan form of a square matrix A, i.e. the "simplest" matrix similar to A. We have that A is diagonalizable if and only if its Jordan form is diagonal.

  `A.jordan_form()` this is gives the Jordan form of A, denoted B

  `A.jordan_form(transformation=True)` this is gives the transition (or similarity matrix), i.e. the invertible matrix P such that A=PBP^(-1). We know that, when A is diagonalizable, the columns of P are linearly independent eigenvectors of A.

- This is how we compute the exponential of a matrix A: `A.exp()`