Produce 1000 points of 2-dimensional normal distribution with a given expected value and given covariance matrix and plot them on the plane. The figure has to be done in two ways:

1. First, with the linear transformation of a random vector variable *X* formed by two independent normal distributed random variables. For this you may use the function normal of the module numpy.random (and e.g numpy.linalg). If M denotes the transformation matrix (then produced points are *MX*, then *(MX)ᵀ(MX)=Xᵀ(MᵀM)X*, thus the covariance matrix is *MᵀM*.)
2. Second, we generate random points by using the multivariate\_normal function of the module numpy.random. One of the arguments of this function is the covariance matrix.

Finally, draw the ellipse that contains most of the dots of the figure. The semi-axes of the ellipse are 6 times the singular values ​​of the transformation matrix (the singular value here coincides with the square root of the eigenvalue of the covariance matrix). For simplicity, the expected value should be 0 and (0,0).

Run the program from the command line. The arguments are the 4 elements of the transformation matrix. For example, the input line python3 6BAdrianSmith.py 2 5 3 0 means that the transformation matrix is

