

**Zadanie 5** – znalezienie wartości własnych i wektorów własnych macierzy hermitowskiej

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

Do rozwiązania tego zadania napisałam program w języku Python z wykorzystaniem biblioteki Numpy.

Macierz przedstawiłam jako sumę dwóch macierzy – rzeczywistej i urojonej. Macierz liczb urojonych utworzyłam jako zwykłą macierz a następnie wywołałam funkcję `imaginary_matrix.astype(complex) * 1j`

Do znalezienia wartości własnych i wektorów własnych wykorzystałam funkcję biblioteczną `linalg.eigh`, która zwraca wartości własne i odpowiadające im znormalizowane wektory własne macierzy hermitowskiej.

## numpy.linalg.eigh

`numpy.linalg.eigh(a, UPLO='L')`

[\[source\]](#)

Return the eigenvalues and eigenvectors of a complex Hermitian (conjugate symmetric) or a real symmetric matrix.

Returns two objects, a 1-D array containing the eigenvalues of *a*, and a 2-D square array or matrix (depending on the input type) of the corresponding eigenvectors (in columns).

Oto uzyskane wyniki:

Eigenvalues:

`[-2.00000000e+00 -4.51028104e-17 0.00000000e+00 2.00000000e+00]`

Eigenvector:

`[-0.5+0.j -0.70710678+0.j 0.+0.j 0.5+0.j]`

Eigenvector:

`[5.00000000e-01+0.j 2.45326947e-17+0.j 0.00000000e+00+0.70710678j 5.00000000e-01+0.j]`

Eigenvector:

`[0.-0.5j 0.+0.70710678j 0.+0.j 0.+0.5j]`

Eigenvector:

`[0.+5.00000000e-01j 0.+2.45326947e-17j 0.70710678+0.00000000e+00j 0.+5.00000000e-01j]`