Python programming practice:
AIMS-Sénégal

Recall some useful functions we have met:
`np.linspace(), np.arange(), np.zeros(), np.ones(),`
`np.random.randn(), np.sum(), np.shape(), len(), plt.plot(), plt.show()`

We will also use some new ones, whose help you can read for usage (and the names are also often suggestive of purpose).:
`np.mean(), np.std(), np.sum(), np.corrcoef(), plt.legend(),`
`plt.savefig(), np.random.randint(), np.nonzero()`

Q1) Make the following numerical arrays (or array equivalents from their mathematical description):

- let a be: a four component vector of all zeros (floats).
- let b be: a vector of 9 True values.
- let c be: a 3×6 matrix of ones.
- $d \in \mathbb{Z}^{2\times4\times3}$.
- let e be: an array of 151 floating point values evenly spaced in the interval [-10, 10].
- let f and g each be: an array of 100 random numbers using `np.random.randn()`
- let h be: an array of 11 random numbers in the interval $[0, 5]$ using `np.random.randint()`

Q2) *i*) Use np.mean() and np.std() to calculate the mean and standard deviation of the random numbers of f and g. Are they they the same? Similar?
ii) Use np.corrcoef() to find the Pearson correlation between f and g. Is it large? What is the format of the output?

Q3) *i*) Use a for loop to print out all the elements of f.
ii) Use a for loop to print out only the *positive* elements of g.

Q4) Use a *for*-loop to make a new array m such that $m_i = f_i\, g_i$, for all i. Use the same loop to calculate the `msum`, the dot product of `f` and `g`.

Q6) Use a *for*-loop to make an array n of length 17, such that $n[i]$ is $+1$ for even i, and -1 for odd i.

Q7) Make a 50×50 array p of a diagonal matrix whose nonzero diagonal values are the square root of the row index. Make the upper half triangle elements be the sum of their indices.