

Università del Piemonte Orientale - DISEI
Business Intelligence for Finance (EC0149)

The following exercises are intended only as examples and do not constitute an exhaustive list of all the possible types of exam exercises.

Exercise 1. (a) Create the function file **funEx1** which takes two vectors u and v of the same length as inputs and computes the following sum by using a **for** loop:

$$\sum_{i=1}^n \left(\frac{1}{u_i} \right)^{v_i}$$

where n is the length of the two vectors. If the vectors do not have the same length an error message is displayed ('vectors do not have the same length').

(b) Create the script file **Exercise1** and, in this file, define the vectors u and v in the following way: u contains the natural numbers from 1 to 10 and v is such that

$$v_i = n - u_i + 1;$$

then use the function **funEx1** to compute the sum of point (a). In the script also define two vectors of different length, and check that the error message is displayed by using **funEx1** on these vectors.

(c) In the same file **Exercise1**, compute the sum of point (a) by using the Matlab predefined function **sum** and the element by element operations.

Exercise 2. Create the script file **Exercise2**.

(a) In this file, define two functions which are the present values of the following projects, through the handle @.

-First project : year 1: +130 ; year 2: +30, year 3: +30.

-Second project: year 1: +50 ; year 2: +50, year 3: +100.

Then, plot the two functions with respect to the discount rate in the interval $[0, 10\%]$ and overlap the graphs by using different colors and markers.

- (b) Through the handle `@`, define the function which is the difference of the two projects. Then, determine the discount rate which makes the two projects equal, by using the function **fzero**.

By observing the plots of the two net present values, which is a good interval that we can pass as an argument to the function **fzero**? Has the function **fzero** given a good result? (Answer by commenting in the file).

Exercise 3. Create the script file **Exercise3**. In this file:

- (a) Generate a vector v with $n=20$ elements, whose i -th component is $v(i) = (-1)^{(i+1)}$. Then, modify the vector in such a way that the last component is equal to 100.
- (b) Generate a vector with $n = 100$ elements of the kind $w = [1, 1/2, \dots, 1/n]$, first with a **for** loop and then by using component by component operations. Check that the two methods give the same result.
- (c) By using the Matlab commands, build the diagonal matrix A , whose principal diagonal contains all the elements from 20 to 2 ($a_{11} = 20$, etc), then extract the submatrix B of dimension 10×10 containing the last ten rows and columns of A . Finally, use the suitable Matlab commands to transform B into the identity matrix. Call C the new matrix thus obtained.

Exercise 4. (a) Create the function file **funEx4** which takes as inputs a square matrix A of dimension $n \times n$ and a vector b of length n , and solves the linear system:

$$A * x = b.$$

By using the command **if**, the function must verify whether the matrix A is invertible and, if not, the function must give an error message (Hint: use a tolerance level of 10^{-8}).

- (b) Create the script **Exercise4**. In the file **Exercise4** solve the two following linear systems by using the function **funEx4**

$$\begin{cases} 6x_1 + 12x_2 + 9x_3 = 3 \\ 2x_1 + 4x_2 + 3x_3 = 1 \\ 1x_1 + 12x_2 + 8x_3 = 4 \end{cases}$$

$$\begin{cases} 6x_1 + 5x_2 + 7x_3 = 3 \\ 7x_1 + 8x_2 + 12x_3 = 2 \\ 2x_1 + 3x_2 + 4x_3 = 4 \end{cases}$$