TP5- MATLAB Scripts and Functions

Part A. MATLAB scripts: In order to be able to reuse the calculation lines, it is useful to put them in a script. A script is a text file that MATLAB can read and execute. To access it: (1) Open the MATLAB script editor either by clicking on the blank page in the toolbar, or by going to the "File/New/M-file" menu.

Exercise 1: Conditional Statements
1. Write the MATLAB script that requests two values x and y from the user and displays them, swaps their contents, and displays them again.
2. Write the MATLAB script that asks for a number and then displays its sign (positive, negative, or zero).

Exercise 2: Repeating instructions (loops)
1. Write a program that requires two integers a and b and displays the result of the following sum:
$$\sum_{k=1}^{b} k^a$$
2. Write a MATLAB program that takes as input a real $x \in]0.20[$, gives an error message if $x \notin]0.20[$, otherwise calculates and outputs the smallest n such that the sum is greater than x.
$$S_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$$
3. Write a program that calculates the 10th term of the Fibonacci sequence:
$$u_0 = 0, u_1 = 1, u_{n+2} = u_{n+1} + u_n$$

Part B. MATLAB Functions:
There are many predefined functions in MATLAB, but there will inevitably come a time when you want to use a function that is not defined. Fortunately, it is possible to define your own functions and use them exactly like pre-existing functions.

Exercise 3:
1. Write a function *pair* that can tell whether an integer x is even or not;
2. Write the *fonction somme* which calculates the sum of two matrices A and B.
3. Write the *fonction produit* which calculates the product of two matrices A and B.

Exercise 4:
Let the following calculation function have two nested loops:
```
function M=calcul(M)
[n,m]=size(M);
for i=1:n
v=M(i,:);
for j=1:m
M(i,j)=v(m-j+1)
end
end
```
1. Give the value of B after executing the following instructions:
   ```
   >> A=[1 2 3 4;5 6 7 8 ;9 10 11 12];
   >> B=calcul(A)
   ```
2. Derive what this function does.
3. Rewrite the calculation function to obtain the same result using a single loop.

Exercise 5 :

1. Write a MATLAB product function that takes two square matrices of the same dimensions A and B as argument and calculates the product $A{\times}B$, then displays the result to matrix C.

2. Write the MATLAB script which allows you to enter two square matrices of the same dimensions A and B, checks if their size is compatible. If they are not, gives an error message and no output, if they are, calculates $AB+BA$. The script calls the product function.

NB: We refrain from using operations of the type $A{\times}B$ or $A+B$, we must instead use an assignment element by element.

Exercise 6:

1. Write a MATLAB "fact" function that takes a positive integer n as an argument and returns $n!$ as an answer. Knowing that $n!=1{\times}2{\times}3{\times}\ldots{\times}n$.
2. Write a MATLAB script which allows you to read two numbers n and p and which calculates and displays

$$n!/(p! * (n-p)!)$$