MAT116 – Programming with MatLab

Exam - 15th February 2020

Lukas Burch, Elise Le Meledo, and Fatemeh Nassajianmojarrad

Institute of Mathematics, University of Zurich

Exercise 1 (1 P.) Series!

1. Write a script called `'series.m'` that:
   - given the series $\sum_{k=1}^{\infty} a_k = \sum_{k=1}^{\infty} [\frac{1}{k} - \frac{1}{k+2}]$, finds the first 10 values of the partial summation $A_n = \sum_{k=1}^{n} [\frac{1}{k} - \frac{1}{k+2}]$ for $n = 1, \ldots, 10$;
   - plots these values as discrete points (no lines between them);
   - adds a title on the top of the plot;
   - adds a label for the x axis;
   - saves automatically the plot in `'series_plot.fig'`;
   - closes the figure.
2. The series $\sum_{k=0}^{\infty} a_k$ converges to $A = \frac{3}{2}$. Build a function `'tolerance_series.m'` that takes as input a tolerance ε and stops the partial sums A_n when the error $|A_n - A| \leq \varepsilon$. The output is the first n such that $|A_n - A| \leq \varepsilon$.
   Input: tolerance.
   Output: first n such that $|A_n - A| \leq \varepsilon$.

Exercise 2 (1 P.) How many days old are you?

Create a function called `'age.m'` that, given as input a birth date in format 'dd-Mmm-yyyy' (ex. '13-Mar-1975'), returns the present age of this person.
Input: char array: a date in format 'dd-Mmm-yyyy' (ex. '13-Mar-1975')
Output: integers: `x`, `y` and `z`: numbers that indicate how many years, months and days old is this person, respectively.
Hint: to know what day is today you can use `floor(now)`.

Exercise 3 (2 P.) Functions (I)

Consider the functions: $f_k\colon [a,b] \to \mathbb{R}$ defined by $f_k(x) \coloneqq x^2 + \dfrac{\sin(k\pi x)}{x+1}, \quad k \in \mathbb{N}$.

1. Write a matlab function called 'values.m' that takes as inputs the variables a, b, k, n and computes as output the **column**-vector containing the values of the function evaluated at the equispaced points:
$$x_i \coloneqq a + \frac{(b-a)}{n-1} \cdot i \quad \text{for } i = 0, 1, \ldots, n-1.$$
Input: a, b, doubles
k, n, integer
Output: y, column-vector of doubles.
Hint: your function should work like: $[y] = values(a, b, k, n)$;

2. Write a script called 'visualisation.m' that plots $(x_i, f_k(x_i))$ for $k \in \{1, 3, 5\}$ and $a = 0, b = 2, n = 1000$ in the same figure and:
   - adds a legend on the top-right of the figure;
   - saves automatically the figure in 'visualisation_figure.fig';
   - closes the figure.

3. Write a function called 'weighted_sum.m' that computes $z = \dfrac{k}{n} \displaystyle\sum_{i=0}^{n-1} \sqrt[3]{i+1}\, f_k(x_i)$ for any given set of parameters a, b, k, n.
Input: same as before.
Output: z, double.
Hint: Your function should work like: $[z] = weighted\_sum(a, b, k, n)$;
You might want to use the function 'values' from the first part.

Exercise 4 (2 P.) Functions (II)

Consider the function: $f\colon [0, \pi] \to \mathbb{R}$ defined by $f(x) \coloneqq \sin(x)$.

1. Write a script called 'periodic.m' that:
   - plots the function on the interval $[0, \pi]$;
   - names the x-axis as 'time [t] = s';
   - saves the plot in 'periodic.fig';
   - closes the figure.

2. Write a function called 'montecarlo.m' that, given as input an integer $n \geq 1$, calculates as an output
$$I = \frac{\pi}{n} \sum_{i=1}^{n} f(x_i),$$
$$E = \left| \frac{\pi}{n} \sum_{i=1}^{n} f(x_i) - 2 \right|,$$
where x_i are uniformly distributed random values in the interval $[0, \pi]$.
Input: n, integer
Output: I, double.
E, double.
Hint: your function should work like: $[I, E] = montecarlo(n)$;
You might want to use the commands: `mean` and `rand`.

3. What do you observe for $n \to \infty$ (try out large values of n, and type your answer in the function description).

Exercise 5 (2 P.) Mistery

1. Create a function called `'mistery'` that:
   - loads the structure `'mistery_content.mat'`, containing a matrix of 2D points coordinates called `mistery_points`, the first column giving the x-coordinate and the second column giving the y-coordinate
   - returns the number N of points which are within the circle of centre (2, 2) and radius 2, boundary included (the number of points satisfying $\sqrt{(x-2)^2+(y-2)^2} \leq 2$), the number of points Px that are on the x-axis ($y = 0$) and number of points Py that are on the $y - axis$ ($x = 0$)

   Input: None
   Output: `N` , integer: the number of points inside the circle
   `Px`, integer: the number of points lying on the x-axis
   `Py`, integer: the number of points lying on the y-axis

2. Create a script called `'mistery_plot.m'` that
   - clears the workspace at the beginning
   - loads the file `'mistery_content.mat'`
   - reshapes the matrix contained in `MisteryMat` into another matrix with dimensions 108×91, stored in the variable `ReshapedMat` and saves it to `SpyMatrix.mat`
   - `spy` the matrix `ReshapedMat`
   - saves the plot into `'mistery_solved.fig'`
   - closes the figures

Exercise 6 (2 P.) Tips and trucks

A rural city wants to limit the amount of trucks and heavy car passing by their inner town. To this aim, they decided to rebuild the main road in a lighter structure so that heavy cars cannot pass anymore. They are asking for your help in order to determine the maximum weight that the road should support so that the car traffic drops by 50%.

1. In order to determine the road's properties, create a script `structure_road.m` that
   a) loads the file "Cars.mat" giving information about cars passing through the town
   b) selects the maximum weight the road should support to allow only 50% of the cars to pass by the inner town and store the value in the variable `MaximumWeight`
   Hint: Pay attention to the type of the variable `Cars` and think of `str2double`

2. Let's now filter the cars going through the town.
   a) Create a function `GoThrough` that given a car weight, tells if it can cross the village

   Input: `W`, float: the actual car weight
   `MaximumWeight`, float: the maximum car weight (inclusive)
   Output: `ok`, boolean: true if the car can go, false otherwise

b) The big sanitary vehicles, that are the only ones of "Chevrolet" brand should always be able to pass through the city and were granted an exception. Create another function `GoThroughPrioritary` that takes the car weight and the car brand as inputs, and tells if a car can go through the city considering the weight constraint and the exception case

Input: `W`, float: the actual car weight
`Brand`, string: the car's brand
`MaximumWeight`, float: the maximum car weight (inclusive)

Output: `ok`, boolean: true if the car can go, false otherwise
`prio`, boolean: true if the car is a sanitary vehicle, false otherwise

Hint: think of `strfind`, and be careful to the case

c) Write a function `test_the_car` that, given the index of the car passing through the city in the given list, tells if the car will be able to pass or not through the city in the future. Use the **exact** formatted sentence "The car number 125 can pass through." or "The car number 125 can not pass through." by replacing 125 by the actual car number. Return the sentence in the variable `OutputSentence`.

Input: `Id`, integer: the index of the car in the given list
`MaximumWeight`, float: the maximum car weight (inclusive)

Output: `ok`, boolean: true if the car can go, false otherwise
`OutputSentence`, string: formatted sentence telling if a car can go or not.

Hints: Your function should load again "Cars.mat". You can use the function `GoThroughPrioritary` inside.