

MATH 131: Numerical Methods for scientists and engineers - Assignment 1

Homework Assignment 1 due by 11:59 PM, February 1, 2021. Test your answers in Matlab.

**For this assignment you are allowed *five* submissions per exercise.
Save each code as an .m file. We will use those codes later in class.**

Some advice first about Math 131

1. Read the syllabus of this course. Do you have any comments, questions or suggestions?
2. Familiarize yourself with MATLAB.
 - (a) Report on your MATLAB version. Are you planning to use it on your personal computer? If so, is it installed? If not, are you planning to use the app on an iPad/mobile?
 - (b) If MATLAB is new to you, the best way to get started is to read MATLAB's internal documentation via the Help browser. From within MATLAB, you can open the Help browser by typing "doc" at the prompt of the command window. Study the "Getting Started with MATLAB" guide, which is the first item under "MATLAB"; if not already visible, expand the "Contents" tab on the left of the screen by clicking on the little icon above the word "Contents". In this guide read the Introduction, Matrices and Arrays, Graphics and the Desktop Tools and Environment sections.
 - (c) MathWorks(<http://www.mathworks.com/help/matlab/http://www.mathworks.com/help/matlab/>) also has a great documentation available online. Google for instance "plot in matlab" and go to the MathWorks result. Note the "Getting Started with MATLAB" section. How does this section compare with the same section in MATLAB's internal documentation. Browse thorough the "Tutorials" section. Try out a couple of commands from this website.
 - (d) What other resources you found for getting help with MATLAB? For instance, what happens if you type "help plot" in MATLAB's Command Window? Try to get help in the same window for other commands as well.

Exercises submitted to Matlab Grader:

1. (15 pts) Write a MATLAB function, called `mysum` that inputs the integer `n` and outputs the sum of the first `n` positive integers (i.e., $\sum_{i=1}^n i$). If a negative `n` is input, make sure to output the value -1 and print the following statement to the user: 'cannot input a negative number'. Use a for loop to compute the sum - don't use MATLAB's built in summation function. Your function should have a header that looks like:

```
function y=mysum(n)
```

where `n` is the number you stop the sum at and `y` is the returned sum.
Hint: You can check your code in MATLAB (NOT MATLAB GRADER) by typing the following into the command line:

```
myresult = mysum(3)
```

and check that `myresult` is equivalent to `1+2+3`
2. (15 pts) Write a Matlab function, called `myabsolutevalue` that inputs a real number `a` and outputs the absolute value of `a`. Use a if-else statement to return the result - don't use Matlab's built in function. The header to your function should look like:

```
function y=myabsolutevalue(a)
```

where `a` is the number you would like to apply an absolute value to and `y` is the returned `|a|`.
Hint: You can check your code in MATLAB (NOT MATLAB GRADER) by typing the following into the command line:

```
myresult = myabsolutevalue(-3)
```

and check that `myresult` is equivalent to `| - 3 |`

MATH 131: Numerical Methods for scientists and engineers - Assignment 1

3. (15 pts) Write a Matlab function, called `myvectornorm` that computes the 2-norm of a vector $\|\vec{x}\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$. Write a Matlab function, called `myvectornorm` that inputs the vector `x` and outputs the 2-norm of `x`. Do not use built-in Matlab functions to calculate the norm (i.e., `norm` or `vecnorm`). The header to your function should look like:

```
function output_norm=myvectornorm(x)
```

where `x` is the input vector and `output_norm` is the returned norm of `x`.

Hint: You can check your code in MATLAB (NOT MATLAB GRADER) by typing the following into the command line:

```
myresult = myvectornorm([1,0,1])
```

 and check that `myresult` is equivalent to the norm of the vector `[1,0,1]`.

4. (15 pts) Let the sequence a_n be defined by:

$$a_{n+1} = \frac{a_n^2 - n^2}{2}, \quad n > 1 \text{ with } a_1 = 1$$

Write a function called `for_sequence` that calculates the n th term of the sequence using a for loop. Your function should have one argument, `n`. The header to your function should look like:

```
function y=for_sequence(n)
```

where `n` is the number of the term in the sequence you are looking for and `y` is the returned term, a_n .

Hint: You can check your code in MATLAB (NOT MATLAB GRADER) by typing the following into the command line:

```
myresult = for_sequence(2)
```

and check that `myresult` is the correct value for a_2 . Additionally, you can check your output from the function `recursive_sequence` if you have already completed it.

5. (15 pts) Let the sequence a_n be defined by:

$$a_{n+1} = \frac{a_n^2 - n^2}{2}, \quad n > 1 \text{ with } a_1 = 1$$

Write a function called `recursive_sequence` that calculates the n th term of the sequence using a recursive function (see the readings for more details). Your function should have one argument, `n`. The header to your function should look like:

```
function y=recursive_sequence(n)
```

where `n` is the number of the term in the sequence you are looking for and `y` is the returned term, a_n .

Hint: You can check your output compared with the other written function `for_sequence` if you have already completed it.

6. (15 pts) Create a scatter plot with a line through it: Scatter plot the data points `datx = rand(20,1);` and `daty = rand(20,1);`. Next, plot the function $f(x) = x$ on the interval where the left endpoint and the right endpoint are the minimum and maximum of `datx`: `x = [minimum: increment: maximum]`. Let the increment = 0.01. The scatter plot and the function plot must be on the same figure. Make sure to label your axes, add a legend to your plot where the legend box appears in the position 'best'. Make sure to define `y = f(x)=x` in your code. (*Hint:* Examine the examples Plot a Function, Plotting two functions, Plotting a line through data points, and Adding a legend in Matlab Grader).

Exercises due on CatCourses:

7. (10 pts) Write any questions you have about the course and/or syllabus and upload as a pdf. If you have no questions, please upload a .pdf that says 'None'. This is partly to make sure that you are able to upload the correct file formats for future assignments. You can use CamScanner or TinyScanner to turn pictures from your phone to a .pdf.