# Recitation 1 (02/03/2016)

**Recitation Instructor**: Shivam Verma
**Email**: shivamverma@nyu.edu
**Ph**: 718-362-7836
**Office hours**: TBD

## How to get MATLAB (5 min)

- Remote access via Courant (see http://cims.nyu.edu/webapps/content/systems/software/purchased) - you will need a CIMS account
- VCL Login (go to home.nyu.edu, click on “VCL Login” button at bottom-left, and follow instructions. You should be able to use MATLAB.)
- Purchase MATLAB as a student - 50$ (see https://www.mathworks.com/store/link/products/student/new?s_tid=ac_buy_sv_cta)
- Use Octave instead (see https://www.gnu.org/software/octave/download.html)

## MATLAB Basics (20 min)

1. Datatypes - int, float, string, vector, matrix, cell
2. Basic arithmetic
3. Bitwise operators - &&, ||, ~=, ==, <=, >=, <, > etc.
4. Creating a variable (with ‘;’ and without)
5. Basic functions - log, log10, exp, sin, cos, sinh, sqrt
6. Vectors - definition, row vector, column vector, linspace, vector from range (eg. 1:0.1:10), accessing vector elements
7. Basic vector algebra - addition/subtraction of vectors, multiplication by constant, dot product, element-wise operation, transpose

8. Matrices - definition, zero matrix, identity matrix, addition/subtraction of matrices, matrix multiplication, accessing matrix elements
9. Common linear algebra operations - matrix-vector multiplication, matrix transpose, inverse, determinant, eigen values, trace, dot,
10. Other operations - matrix/vector concatenation (horizontal/vertical), common functions applied to vector/matrix - exp, log, sqrt, max, min, mean, size, length.
11. Go through this brief MATLAB tutorial (recommended for MATLAB beginners) - https://learnxinyminutes.com/docs/matlab/

## Basic commands

- Use CTRL-C (Mac/Windows) to stop an execution/script.
- To know more about a function/command, use the command **help <func_name>.**
- clc - clear screen
- clear <var_name> - clear that variable from namespace
- clear / clear all - clear all variables from namespace
- See “Common MATLAB commands” - http://www.hkn.umn.edu/resources/files/matlab/MatlabCommands.pdf

## Programming constructs in MATLAB (15 min)

1. Reading input: a = input(‘Enter value: ‘);

2. Printing output: disp(a)

3. Conditionals:

```
if (a>10 && a<20)
        disp('a is between 10 and 20')
elseif (a>=20 && a<30)
        disp('a is between 20 and 30')
else
```

```
        disp('a is between 30 and inf')
end;
```

4. For loop:

```
for k=1:2:10
        disp(k)
end;
```

5. While loop:

```
k=0;
while(k<5)
        k=k+1
end;
```

6. Writing a function/closure

```
function vec = AddTwoVecs(a,b)
        vec = zeros(size(a));
        if (isvector(a) && length(a)~=length(b))
                disp('Error: Vector sizes do not match');
                return
        else
                for i=1:length(a)
                        vec(i)=a(i)+b(i);
                end;
        end;
end
```

7. Anonymous functions

```
sqr = @(x) x.^2; sqr(5)
```

- sum_sqr = @(x,y) x.^2 + y.^2;
  - sum_sqr(1,3)
  - sum_sqr([1,2],[3,4])

8. Random numbers
    - help rand
    - help randn

## Plotting in MATLAB (10 min)

```
a=0:0.1:10;
f_x = @(x) x*exp(x) - 1;
g_x = @(x) x.^2;

for i=1:length(a)
        fa(i)=f_x(a(i));
        ga(i)=g_x(a(i));
end;
plot(a,fa,'b-') %see help for more options
title('Plotting functions')
ylabel('F(x)');
xlabel('x');
hold on
plot(a,ga,'r-')
legend('x*exp(x) - 1','x^2','Location','Best')
```

## Floating point system in MATLAB (5 min)

- Due to finite precision in floating point number representation, there are gaps between consecutive numbers.
- Size of these gaps depends on the size of the number and on the precision (e.g., double or single precision).
- MATLAB has the function eps(), which returns, for a number, the distance to the next floating point number in the same precision.

Examples:

- eps(1)
- eps(single(1))
- eps(2^(40))
- eps(single(2^(40)))

Try this:

- a = 0.8; b = 0.7;
  - a - b == 0.1 : Can you predict the answer?
  - a - b - 0.1

What do we get above and why? **Find the answer [HW.]!** Read about floating point representations in computing:

- https://en.wikipedia.org/wiki/IEEE_floating_point
- Numerical Computing with IEEE Floating Point Arithmetic, Michael L. Overton (NYU)
- Sec. 2.5, Numerical Mathematics, Alfio Quarteroni et al.
- Sec. 2.1, Numerical Analysis in Modern Scientific Computing, Peter Deuflhard and Andreas Hohmann.

**Hint**: Can you represent 0.8 & 0.7 as dyadic rationals in the floating point system (of form a/(2^n))? Can you represent 0.1?

## Fixed point (iteration) method in MATLAB (15 min)

$f(x) = x * e^x - 1$
Solve: $\phi(x) = x + k \cdot f(x)$ using the fixed point method. Take $x_0 = 0.5, k = -0.40435,$ tolerance = 10e-8.

```
x_0=0;
k= -0.40435;
```

```
f_x=@(x) (x*exp(x)-1);
count=0;
x_k=0.5;

while abs(x_k-x_0)>=10^(-8)
   x_0=x_k;
   x_k=x_0 + k*f_x(x_0);
   count=count+1;
   if (count>500)
      break;
   end
end
```

**Try this at home [HW.]**:

1. Try different values of the parameter 'k' (say 0.1, 0.2, 0.5, 0.7), keeping everything else fixed. What do you observe? Can you explain this behaviour?
2. For different values of 'k' (as above), play around with the initial value $x_0$. What change do you observe in the number of iterations? How would you explain this?
3. Think about how you can implement Newton's method on MATLAB. Remember, the iterative equation for Newton's method is: $x_k = x_{k-1} + \frac{f(x_k)}{f'(x_k)}$.
4. Try solving the above problem using Newton's method on MATLAB. How many iterations do you get? What can you say about the rate of convergence?

**Common MATLAB mistakes (5 min)**

1. Vector/matrix indices start from 1, not 0 (like Python or other languages)
2. We place a DOT (‘.’) in front of a binary operation when we want that operation to be done element-wise.
    - a = [1,2,3]’
        - a.^2 = [2,4,9]’ , a^2 = ERROR
    - A = [1,2; 3,4]; B = [1,0; 0,-1];
        - A.*B = [1,0; -3,4] , A*B = [-1,2; -1,4]

Questions?