Introduction to MATLAB – Step by Step Exercise

Large list of exercise: start doing now!

1 – 35: Basic (variables, GUI, command window, basic plot, for, if, functions)

36 – 40: Medium (functions)

41 – 45: Medium (matrices)

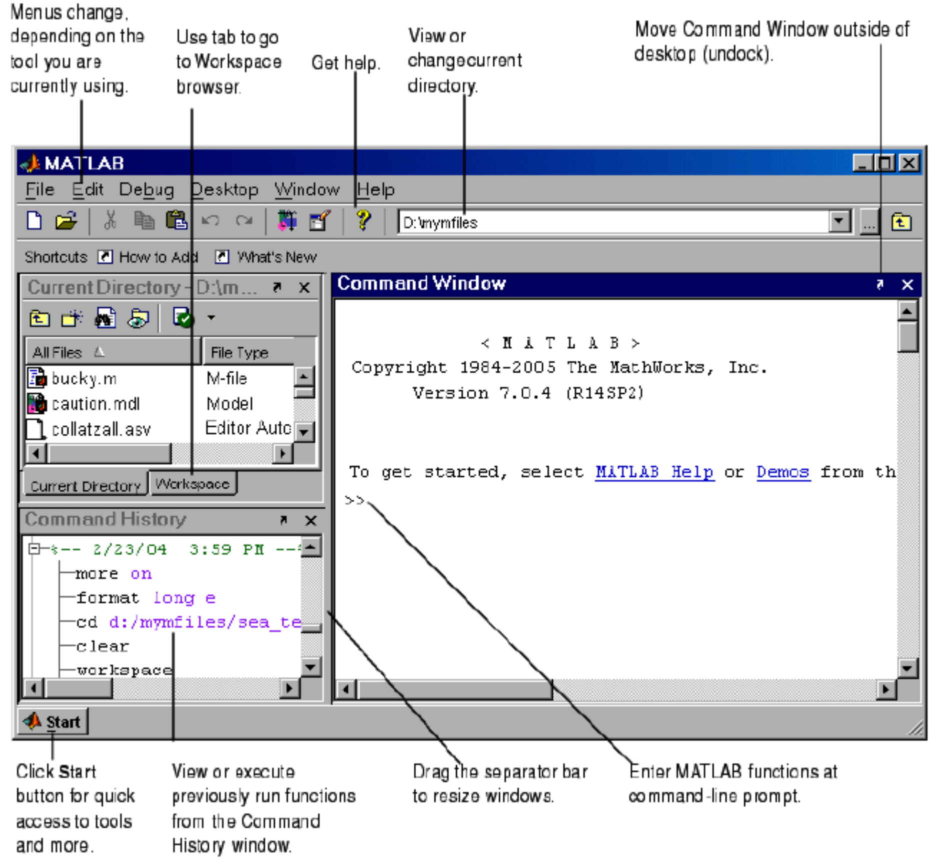
46 – 51: Medium (plot)

52 – 55: Medium (integration)

56 – 60: Advanced (combined problems)

Introduction to MATLAB – Step by Step Exercise

1. Open MATLAB(student AMO/AIR)
2. Make sure that you recognize the Graphic User Interface (GUI)



HØGSKOLEN I ÅLESUND

3. Look for the command window, and use it as a calculator:
   - `2+2`
   - `2 * 2`
   - `22`
   - $$\frac{3\sqrt{3}}{4} + 24\left(\frac{1}{12} - \frac{1}{5\cdot 2^5} - \frac{1}{28\cdot 2^7} - \frac{1}{72\cdot 2^9}\right).$$
   - $$3 + \frac{1}{60}\left(8 + \frac{2\cdot 3}{7\cdot 8\cdot 3}\left(13 + \frac{3\cdot 5}{10\cdot 11\cdot 3}\left(18 + \frac{4\cdot 7}{13\cdot 14\cdot 3}\right)\right)\right).$$

4. Create variables at the command window:
   - `a = 2`
   - `b = 3`
   - `a + b`
   - `first_string = 'My name is '`
   - `second_string = 'yournamehere,andpleasedontcopyandpasteit,justwriteyourname,yourownname,thatonethatyourparentsgaveyoumanyyearsago'`
   - `first_string + second_string`

Introduction to MATLAB – Step by Step Exercise

5. Create variables based on other variables:

- `c = a * 2`
- `d = cos(b)`
- `e = c + d`
- `r = 5`
- `A = 2 * pi * r`
- `C = 2 * pi * r`
- `x = 0`
- `curve_f = sin(x) + cos(x/3+1)`

6. Create vectors:

- `vector_1 = [1 2 3 4 5 6 7 8 9 10]`
- `vector_2 = [12 13 14 15 16 17 8767826264]`

7. Operation with vectors:

- `vec_1 = [1 2 3]`
- `vec_2 = [7 8 9]`
- `vec_1 + 10`
- `vec_1 + vec_2`
- `vec_1  - vec_2`
- `times(vec_1, vec_2)`

Introduction to MATLAB – Step by Step Exercise

8. Create column vectors

- `colu_1 = [1; 2; 3; 4; 5]`
- `colu_2 = [23; 24; 25; 26]`
- `colu_3 = ['aa'; 'bb'; 'cc'; 'dd']`

9. Other ways to create vectors:

- `z = zeros(5,1)`
- `zz = zeros (1, 5)`
- `zzz = [0: 1:10]`
- `zzzz = [-8763: 430.2265 : 5634.23]`

10. Creating Matrices:

- `matr_1 = [1 2 3; 4 5 6; 7 8 10]`
- `matr_2 = ['lala ' 'lele '; 'lili ' 'lolo '; 'lulu ' 'lålå ']`

11. Operation with Matrices:

- `matr_1 + 10`
- `sin(matr_1)`
- `matr_1'`
- `inv(matr_1)`
- `identity_matrix = matr_1 * inv(matr_1)`
- `element_multiplication = matr_1.*matr_1`

12. Accessing elements in the Matrix:

- `matr_1(1,2)`
- `matr_1(8)`
- `matr_1(1:3,2)`
- `matr_1(3,:)`

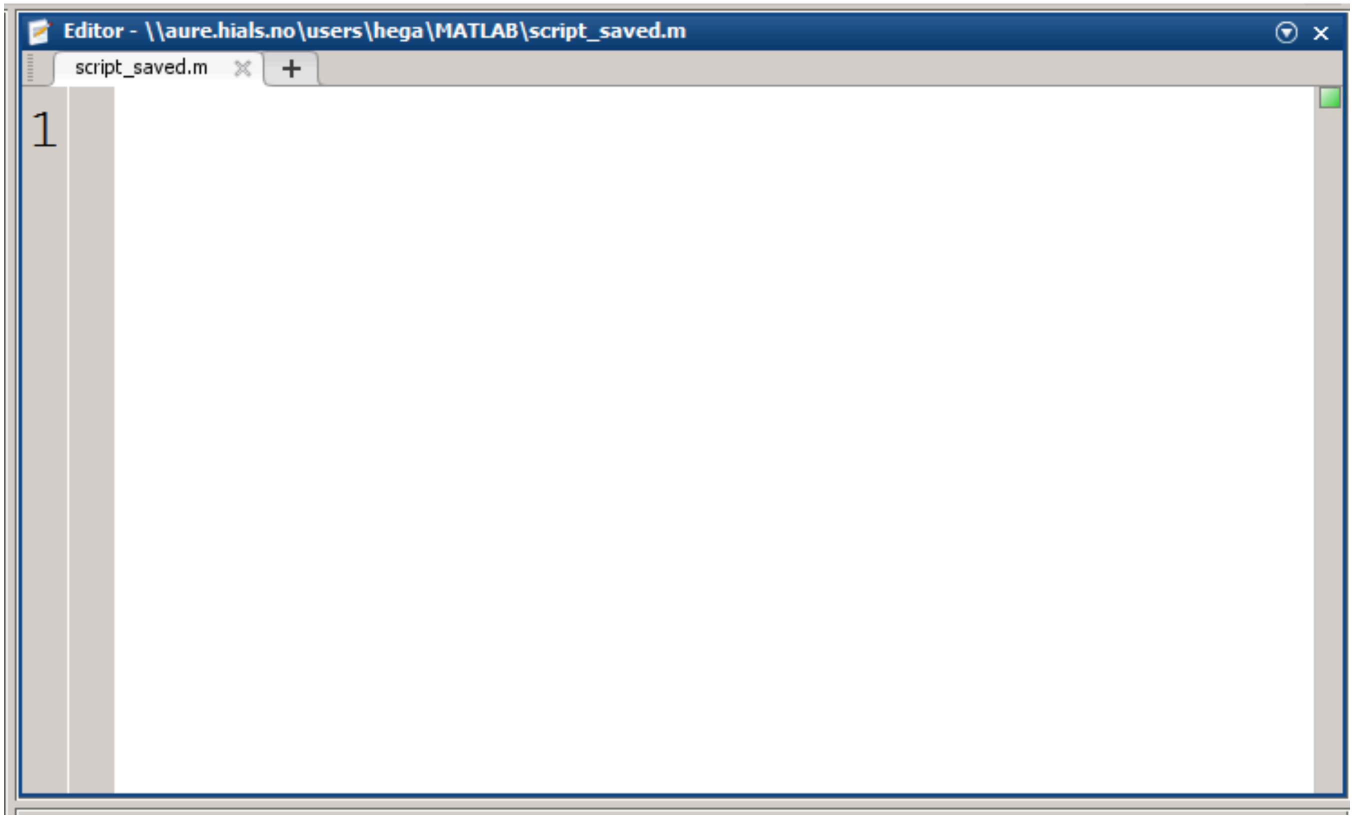
13. Check that your variables are at the workspace:

Workspace

Name	Value	Min	Max
a	[1 2 3;4 5 6;7 8 10]	1	10
A	[1 2 3;4 5 6;7 8 10]	1	10
ans	[2;5;8]	2	8
b	[1;2;3]	1	3
e	3	3	3
first_stri...	'My name is '		
matr_1	[1 2 3;4 5 6;7 8 10]	1	10
matr_2	*3x10 char*		
x	*1x11 double*	0	10
z	[0;0;0;0;0]	0	0
zzzz	*1x34 double*	-8...	5....

Introduction to MATLAB – Step by Step Exercise

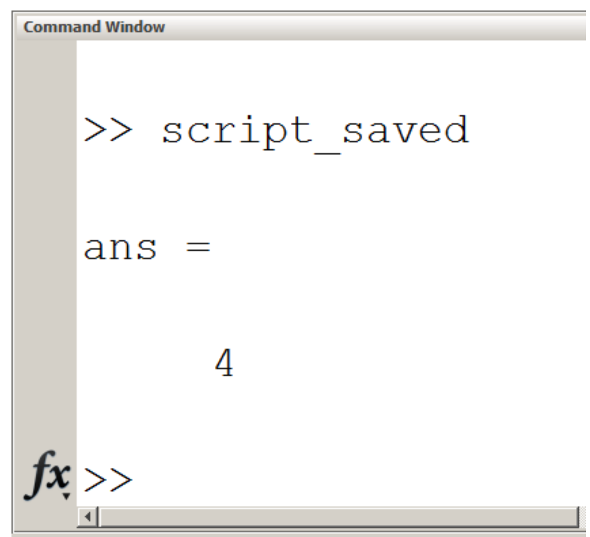
14. Create and save a script (no spaces, MATLAB folder):



14. Start your script by clearing the variables ans summing 2 + 2:

```
1. clear
2. 2 +2
```

15. Run your script and check the answer (ans) on the command window:



16. Create a vector in your script with a list of dates:

```
1. clear
2. dates = [1015 1066 1660 1814 1905 2014]
```

17. Realize that, by putting ; at the end of the line the command does not appear at the command window:

```
1. clear;
2. dates = [1015 1066 1660 1814 1905 2014];
```

18. Sum up all the ages:

```
1. clear
2. dates = [1015 1066 1660 1814 1905 2014];
3. sum_all = sum(ages);
```

19. Save the number of dates inside the vector "dates" into a variable "":

```
1. clear;
2. dates = [1015 1066 1660 1814 1905 2014];
3. sum_all = sum(dates);
4. how_may_dates = length(dates);
```

20. Write a comment

```
5. % This is a comment
6. % Realize that from now the code is your own, so you don't need to follow the same line that I write here.
```

21. Calculate the average of the dates by dividing the sum by the number of elements

```
average_dates = sum_all/how_may_dates;
```

22. Display in the command line a text, and later the average

```
disp('The average is: ');
disp(average_dates )
```

Introduction to MATLAB – Step by Step Exercise

23. Plot the sin(dates)

```
f_x = sin(dates);
plot (dates, f_x);
```

24. Plot $(\text{dates})^2 / (150000) - 0.02 * (\text{dates}) + 12$:

```
ff_x = (dates).^2/(150000) - 0.02*(dates) +12
plot (dates, ff_x);
```

25. Use "hold on" between the two plots :

```
ff_x = (dates).^2/(150000) - 0.02*(dates) +12;
plot (dates, ff_x);
hold on
f_x = sin(dates);
plot (dates, f_x);
```

26. Realize that we can transform numbers to string and use it to display test inside a "disp" as a vector

```
disp(['Dois mais Dois igual a: ' num2str(4)]);
```

27. Create a for to read each element of the vector and display its value

```
for i = 1:how_may_dates
 disp(['The date is: ' num2str(dates(i))]);
end
```

28. Create a "if" to check if a year is before, equal or after year 1800

```
year = 1750;
if year < 1800
   disp('Year is before 1800');
elseif year == 1800
   disp('Year is 1800');
else
   disp('Year is above 1800');
end
```

29. Incorporate and modify the "if" inside your "for", to check if a date is before, after or equal 1814

```
for i = 1:how_may_dates
 disp(['The date is: ' num2str(dates(i))]);
   if dates(i) < 1814
         disp('Before 1814');
   elseif dates(i) == 1814
         disp('It is 1814!');
   else
         disp('After 1814');
   end
end
```

29. Incorporate and modify the "if" inside your "for", to check if a date is before, after or equal 1814

```
for i = 1:how_may_dates
 disp(['The date is: ' num2str(dates(i))]);
   if dates(i) < 1814
         disp('Before 1814');
   elseif dates(i) == 1814
         disp('It is 1814!');
   else
         disp('After 1814');
   end
end
```

30. Adapt your code from 29 to solve the example from last week:

Create a a code that checks if you can buy alcohol in Norway, the type of alcohol, if you can enter in a night club, and if you can teach your friend to drive:

- age < 18 – None
- 18 < age < 20 – Alcohol below 22%, no clubbing nor teach
- 20 < age < 21 Alcohol above 22%, but no clubbing nor teaching
- 21 < age < 25 – Alcohol above 22% and clubbing, but no teaching
- age > 25 – All allowed

Introduction to MATLAB – Step by Step Exercise

31. Function: a named section of a program that performs a specific task. Realized that "sum", "length" and "times" is a function

```
sum([1 2])
length([1 2])
times([2],[2])
```

32. Study the basic command to create a function:

- function to add any two numbers:

function to create a function

what the function returns

what the function receives

```
function [sum_number] = add_numbers(x,y)
    sum_number = x+y;
end
```

name of the function

variable that receives the operation

operation/task performed by the function

Introduction to MATLAB – Step by Step Exercise

33. Based on 32, created a function that adds two numbers called "add_numbers".

34. Use your "add_numbers:

```
add_numbers(2,3)
add_numbers(10,32)
```

35. Create a new function, that multiply 2 numbers, and use it

36. Create a function that transform years in days

37. Create a function that check if a number is above or bellow 1814

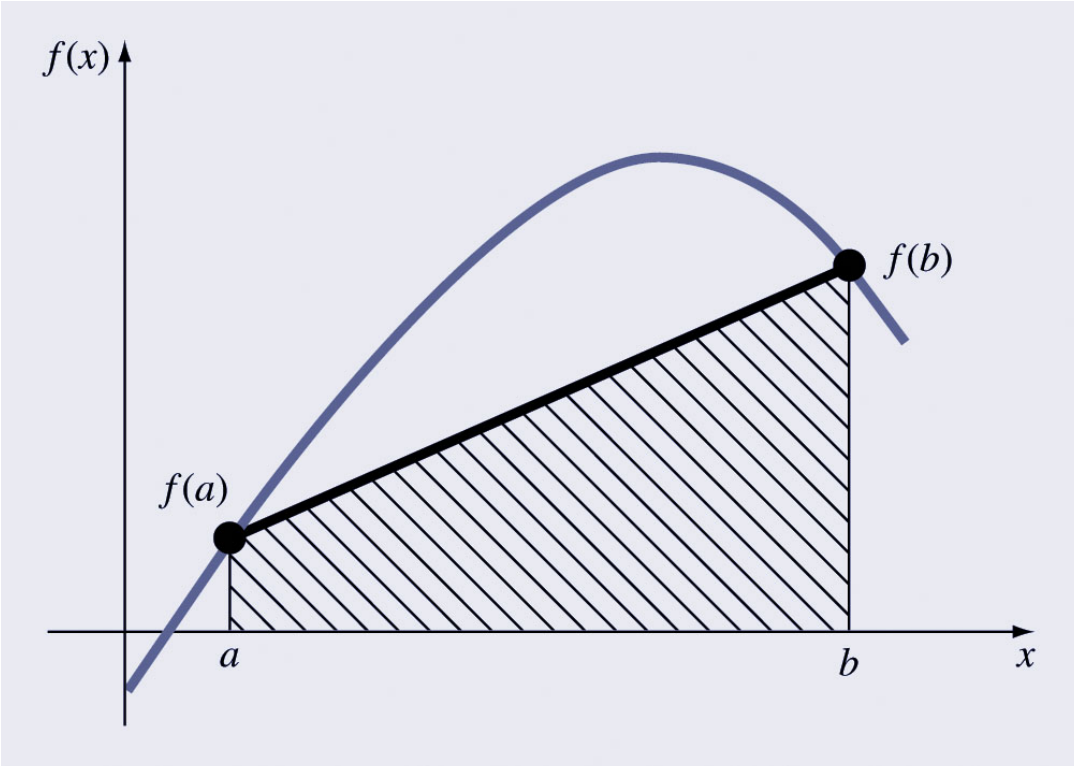
38. Create a function that receives a vector and display all the elements of this vector

39. Create a function that calculates sigma for a cantilever given your P, L and h

```
function [sigma] = tension(P,L,h)
        sigma = P*L*6/(h^3);
end
```

Introduction to MATLAB – Step by Step Exercise

40. Create a function calculate the area (I) between two points (a,b) by the trapezoidal rule:



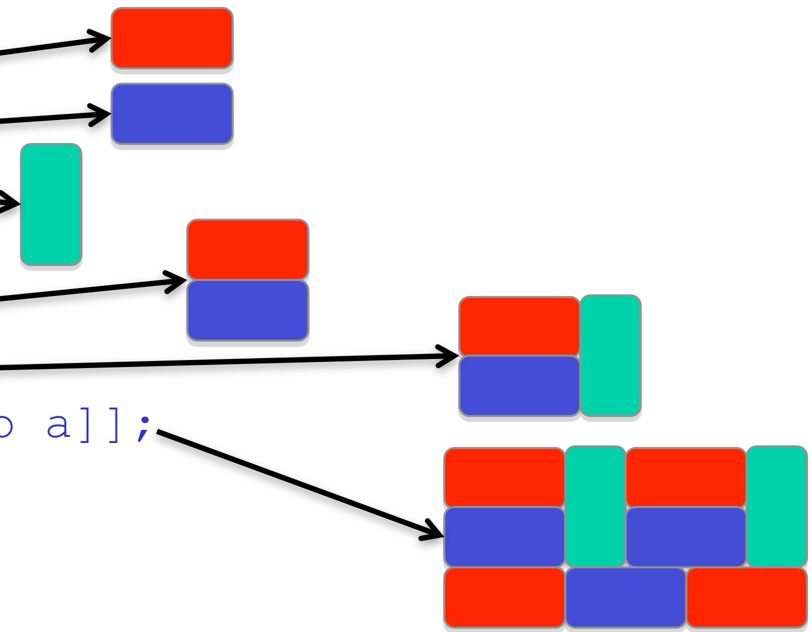
$$I = (b-a)\frac{f(a)+f(b)}{2}$$

HØGSKOLEN I ÅLESUND

Introduction to MATLAB – Step by Step Exercise

41. Create matrices *d*, *e* and *f* by concatenating vectors *a*, *b* and *c*:

```
>> a = [1 2];
>> b = [3 4];
>> c = [5;6];

>> d = [a;b];
>> e = [d c];
>> f = [[e e];[a b a]];
```



42. Consider the a = 2, b=4, c=6, d=9 and calculate 2A in MATLAB given :

$$\lambda = 2, \quad \mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

$$2\mathbf{A} = 2\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 2\cdot a & 2\cdot b \\ 2\cdot c & 2\cdot d \end{pmatrix} = \begin{pmatrix} a\cdot 2 & b\cdot 2 \\ c\cdot 2 & d\cdot 2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} 2 = \mathbf{A}2.$$

43. Consider θ = pi/6, m'=4, n'=2, calculate the value of [m,n] for:

$$\begin{bmatrix} m \\ n \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} m' \\ n' \end{bmatrix}$$

44. Solve the problem from 1st day, calculating how much sales the shop makes on each day in matrix operations:

Matrix multiplication example:

- Beef pies cost \$3 each
- Chicken pies cost \$4 each
- Vegetable pies cost \$2 each

They are sold in 4 days

	Mon	Tue	Wed	Thu
Beef	**13**	**9**	**7**	**15**
Chicken	**8**	**7**	**4**	**6**
Vegetable	**6**	**4**	**0**	**3**

the **value of sales** for Monday is calculated as:

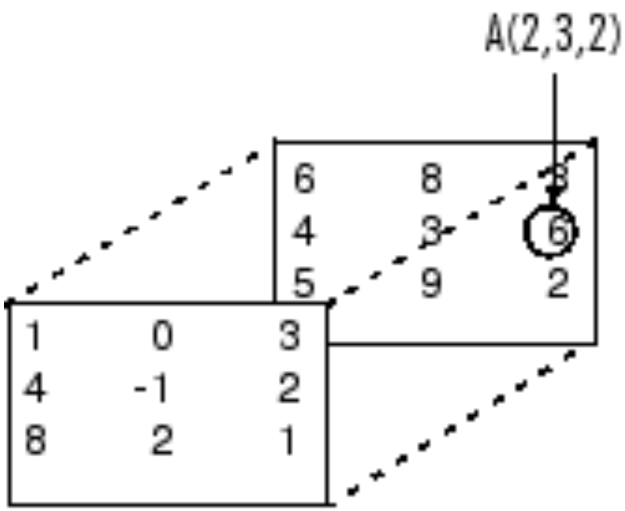
- Beef pie value + Chicken pie value + Vegetable pie value

= \$3×13 + \$4×8 + \$2×6 = \$83

= (\$3, \$4, \$2) • (13, 8, 6) = \$3×13 + \$4×8 + \$2×6 = \$83

Introduction to MATLAB – Step by Step Exercise

45. Create a multi-dimensional matrix based on the figure below:

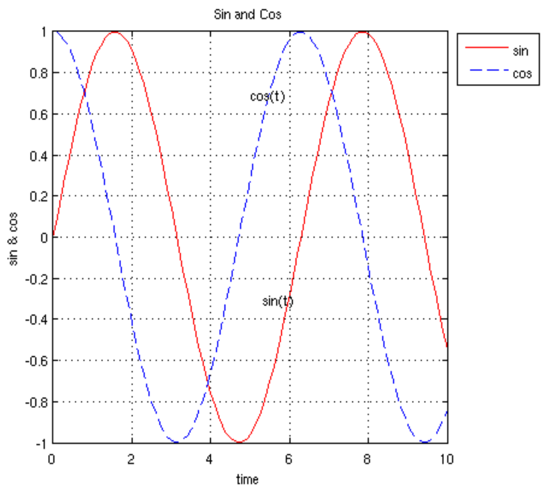



```
A(:,:,1) =

     1     0     3
     4    -1     2
     8     2     1


A(:,:,2) =

     6     8     3
     4     3     6
     5     9     2
```

Introduction to MATLAB – Step by Step Exercise

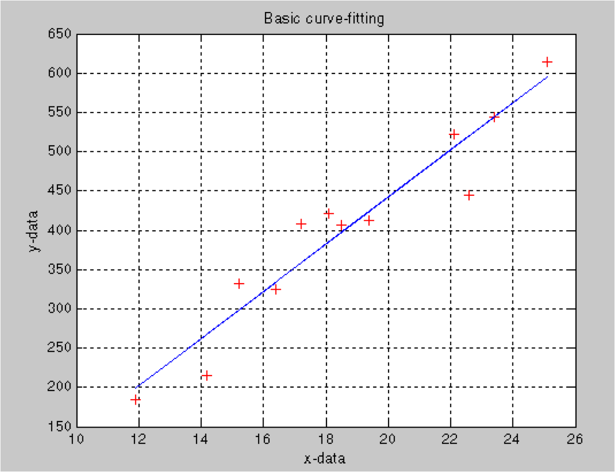
46. Obtain the following plot:

```
t=0:0.1:10;
y1=sin(t);
y2=cos(t);
plot(t,y1,'r',t,y2,'b--');
x=[1.7*pi;1.6*pi];
y=[-0.3; 0.7];
s=['sin(t)';'cos(t)'];
text(x, y, s);   % Add comment at (x,y)
title('Sin and Cos');   % Title
legend('sin','cos') % Add legend
xlabel('time')   % the name of X-axis
ylabel('sin & cos') % the name of Y-axis
grid on         % Add grid
axis square        % set figure as a shape
   of square
```



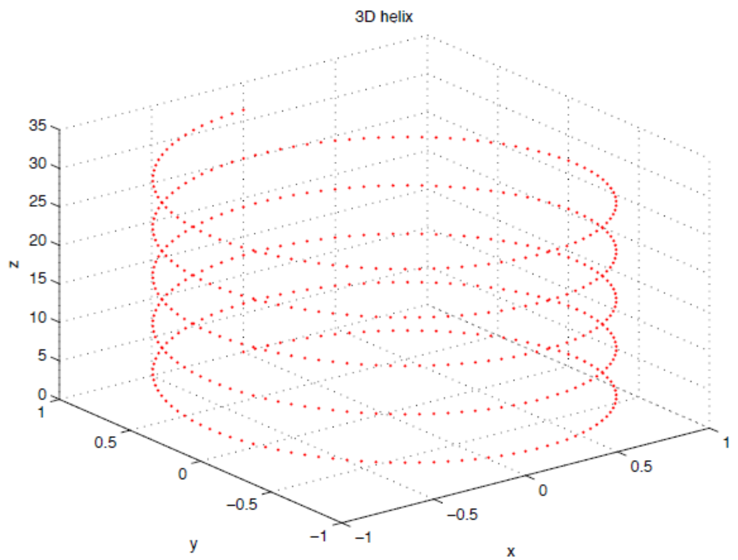
47. Obtain the similar curving fit data using polyfit and polyval:

```
x=[14.2, 16.4, 11.9, 15.2, 18.5, 22.1,
   19.4, 25.1, 23.4, 18.1, 22.6,
   17.2];
y=[215, 325, 185, 332, 406, 522, 412,
   614, 544, 421, 445, 408];

coeff = polyfit(x,y,1);
y_fit = polyval(coeff,x);

plot(x,y,'r+',x,y_fit), grid on,
   xlabel('x-data'), ylabel('y-data'),
   title('Basic curve-fitting'),
   legend('Original data','Line of
   best fit','Location','SouthEast')
```



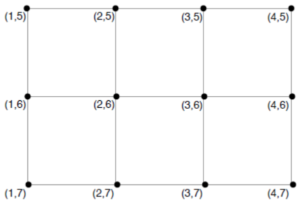
48. Obtain the following 3D plot:

```
t=0:pi/50:10*pi;
plot3(sin(t),cos(t),t, 'r.'),grid
on,xlabel('x'),
ylabel('y'),zlabel('z'),
title('3D helix')
```



Introduction to MATLAB – Step by Step Exercise

49. Define a meshgrid and plot the following 3D function:



$$z = c \cdot \sin\left(2\pi a\sqrt{x^2+y^2}\right),$$

where a = 3, c = 0.5, -1 < x < 1 and -1 < y < 1

```
x=linspace(-1,1,50);
y=x;
a=3
c=0.5
[xx, yy] = meshgrid(x,y);
z = c*sin(2*pi*a*sqrt(xx.^2+yy.^2));
surf(xx,yy,z), colorbar, xlabel('x'), ylabel('y'),
zlabel('z'),title('f(x,y)=c sin(2 \pi a \surd(x^2+y^2))')
figure;
mesh(xx,yy,z), colorbar, xlabel('x'), ylabel('y'),
zlabel('z'), title('f(x,y)=c sin(2 \pi a \surd(x^2+y^2))')
```

50 Plot the following 3D curves using the **plot3** function

a) Spherical helix

$$x = \sin\left(\frac{t}{2c}\right)\cos(t)$$
$$y = \sin\left(\frac{t}{2c}\right)\sin(t)$$
$$z = \cos\left(\frac{t}{2c}\right)$$

where c = 5 and 0 < t < 10π

b) Sine wave on a sphere

$$x = \cos(t)\sqrt{b^2 - c^2\cos^2(at)}$$
$$y = \sin(t)\sqrt{b^2 - c^2\cos^2(at)}$$
$$z = c \cdot \cos(at)$$

where a = 10, b = 1, c = 0.3, and 0 < t < 2π

51 Plot the following 3D curves using the **surf** function

Sine surface

$$x = \sin(u)$$
$$y = \sin(v)$$
$$z = sin(u + v)$$

where 0 < u < 2π and 0 < v < 2π

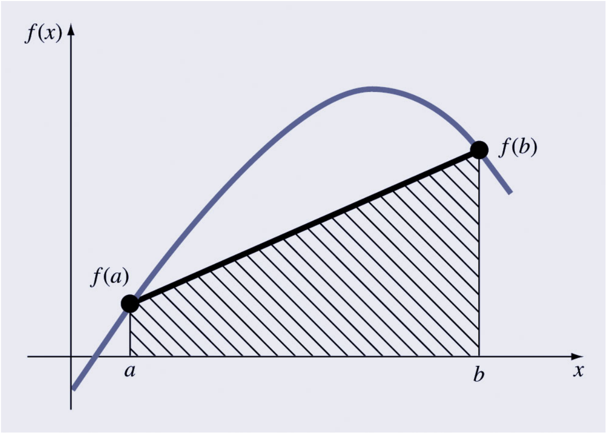
Elliptic torus

$$x = [1 - r_1 \cos(v)] \cos(u)$$
$$y = [1 - r_1 \cos(v)] \sin(u)$$
$$z = r_2 \cdot \left[\sin(v) + \frac{tu}{\pi}\right]$$

where $r_1 = r_2 = 0.5$, $t = 1.5$, 0 < u < 10π and 0 < v < 10π

Introduction to MATLAB – Step by Step Exercise

52. Describe each part from the trapezoidal function from MATLAB



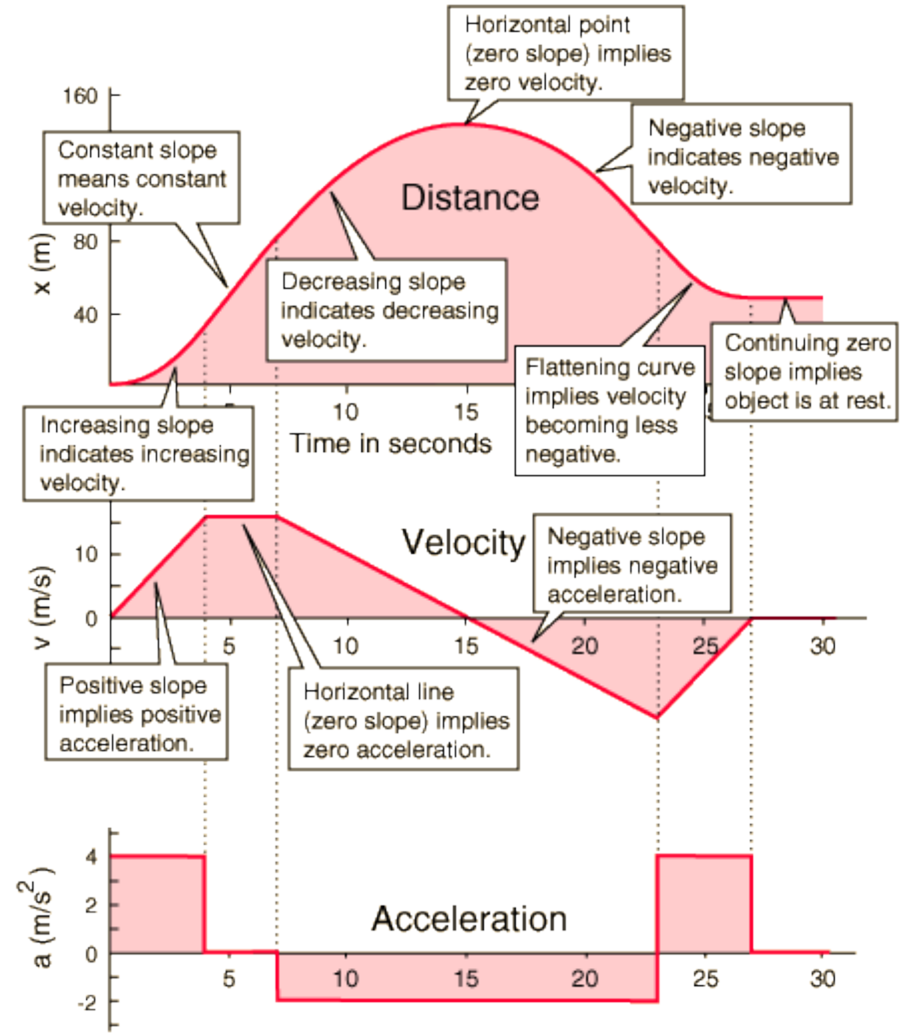

```
function I = trap(func,a,b,n,varargin)
% trap: composite trapezoidal rule quadrature
%   I = trap(func,a,b,n,p1,p2,...):
%                       composite trapezoidal rule
% input:
%   func = function handle to function to be integrated
%   a, b = integration limits
%   n = number of segments (default = 100)
%   p1,p2,... = additional parameters used by func
% output:
%   I = integral estimate

if nargin<3,error('at least 3 input arguments required'),end
if ~(b>a),error('upper bound must be greater than lower'),end
if nargin<4||isempty(n),n=100;end
x = a; h = (b - a)/n;
s=func(a,varargin{:});
for i = 1 : n-1
  x = x + h;
  s = s + 2*func(x,varargin{:});
end
s = s + func(b,varargin{:});
I = (b - a) * s/(2*n);
```

53. Remind about differential equations, and how

$$x(t) = \text{position at time } t$$
$$v(t) = \text{velocity at time } t$$
$$a(t) = \text{acceleration at time } t$$

$$\int_0^T a(t)\,dt = v(T) - v(0)$$

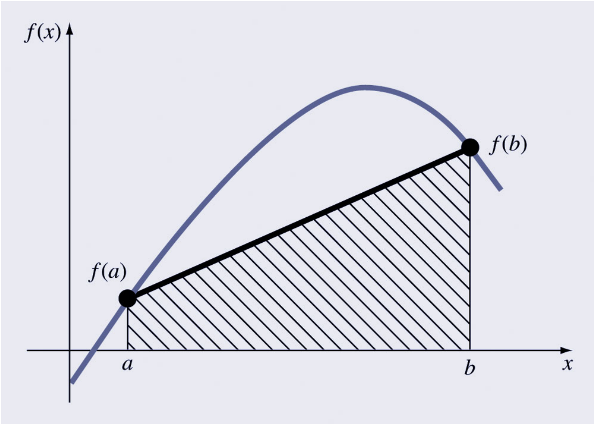
$$\int_0^T v(t)\,dt = x(T) - x(0)$$



Introduction to MATLAB – Step by Step Exercise

54. Using the trapezoidal function plot and integrate (0-pi/2) for f(x) = sen(x) and f(x) = cos(x)

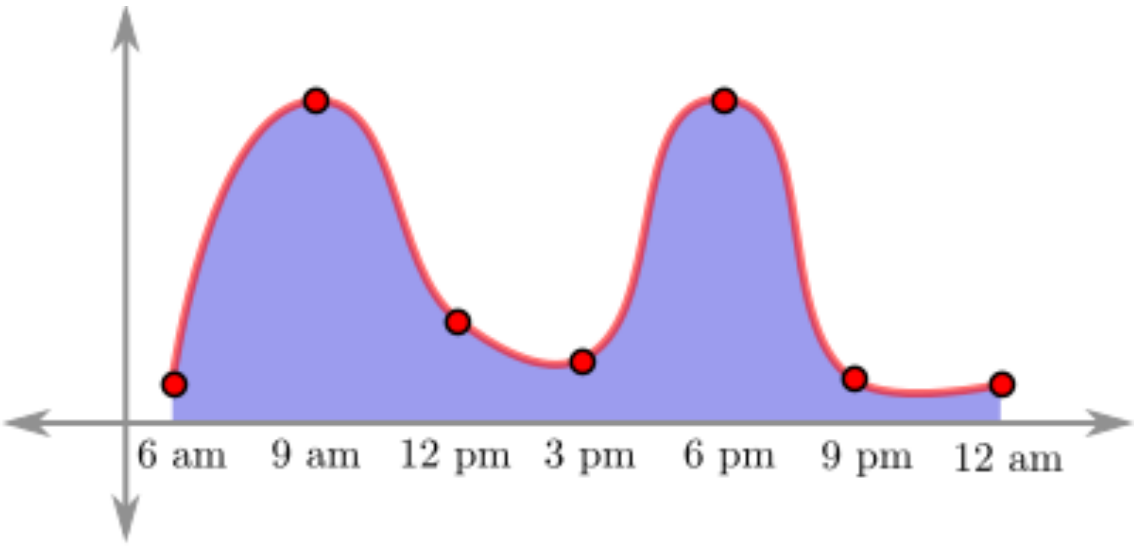
```
x = 0:pi/100:pi;
y = sin(x);
trapz(y,x)  %  returns 1.9338
plot (x,y,'k-*')

%for the lines
for i=1:length(x)
      line([x(i) x(i)], [0 y(i)])
end
```



HØGSKOLEN I ÅLESUND

55. Using the trapezoidal function plot and integrate the number of passengers

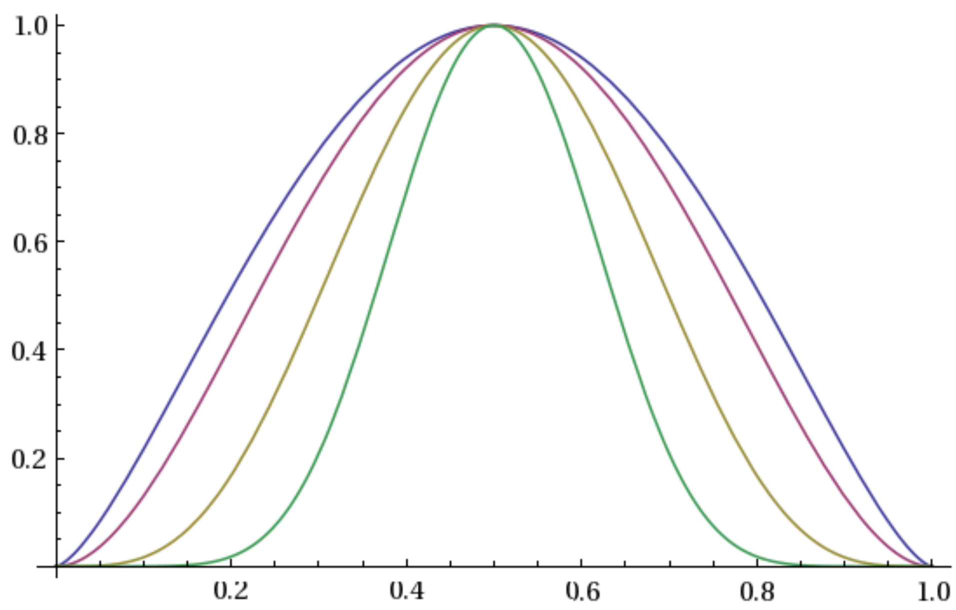
Time	6 am	9 am	12 pm	3 pm	6 pm	9 pm	12 am
Passengers per hour	120	1200	360	240	1200	180	120



56. Plot the bell-shaped function f(x), x range [0,1], varying `α in [1.5, 2, 4, 9 ]`

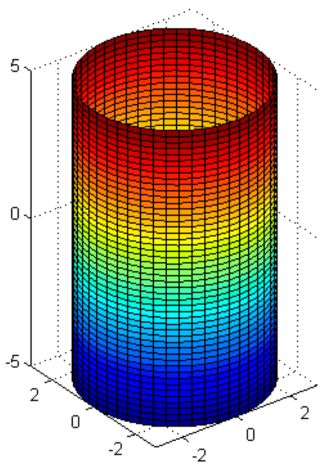
Using the trapezoidal function, calculate the area from the range x [0.2, 0.8] for all four `α`

```
f(x) = 4^α * x^(α - 1) * (1 - x)^(α - 1)
```



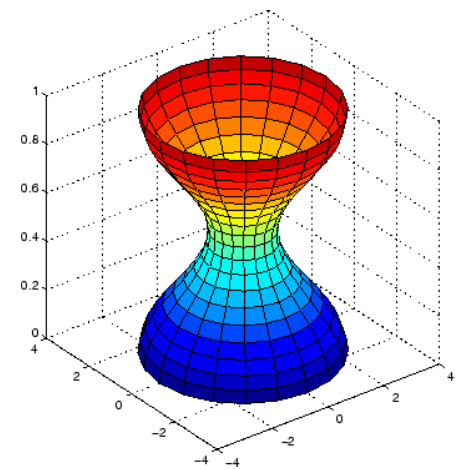
57. Plot the following solids in revolution (cylinder) function and calculate its volume $V = \pi \int_a^b f^2(x)\,dx$

a)




```
A = meshgrid(linspace(0, 2*pi, 50),
linspace(0, 2*pi, 50)) ;
X = 3 .* cos(A);
Y = 3 .* sin(A);
Z = meshgrid(linspace(-5, 5, 50),
linspace(-5, 5, 50))';
surf(X, Y, Z), axis equal
```

b)




```
t = 0:pi/10:2*pi;
[X,Y,Z] =
cylinder(2+cos(t));
surf(X,Y,Z)
axis square
```

HØGSKOLEN I ÅLESUND

Introduction to MATLAB – Step by Step Exercise

58. Design a group of cranes, varying square cross section and load for L = 3m. Check if crane collapses (σ_{max} = 250MPa)

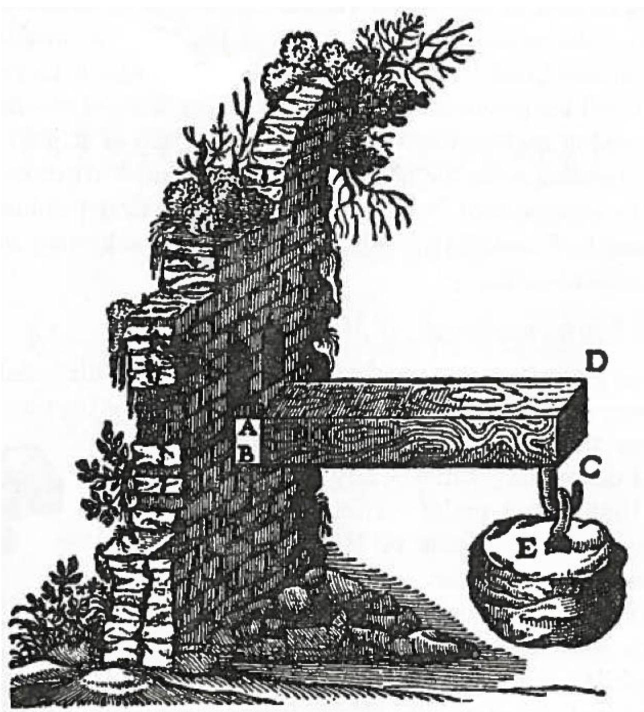
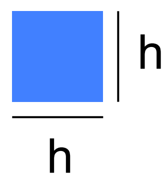
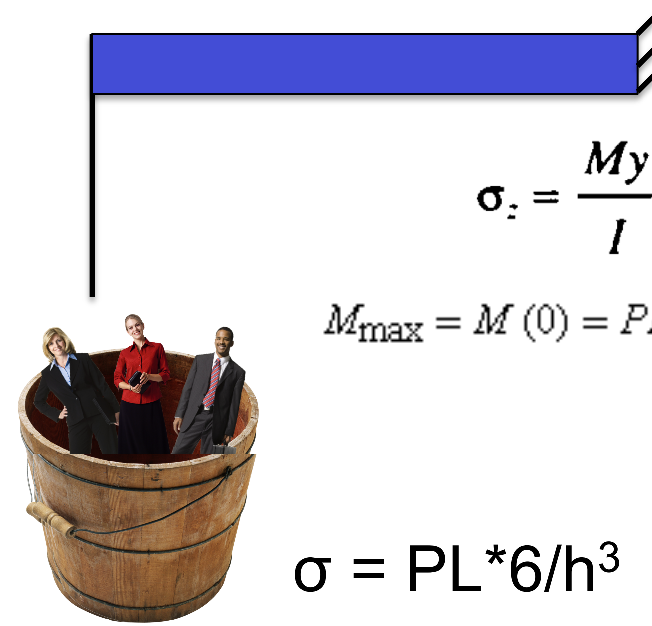


Fig. 15. Galileo's illustration of bending test.





$$\sigma_z = \frac{My}{I}$$

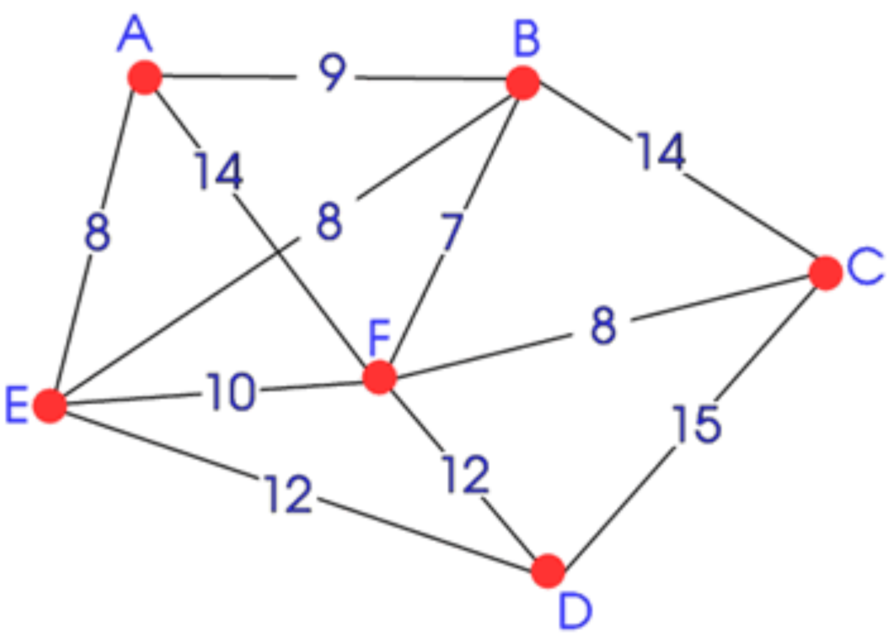
$$M_{\max} = M(0) = PL$$

$$\sigma = PL*6/h^3$$

Consider:

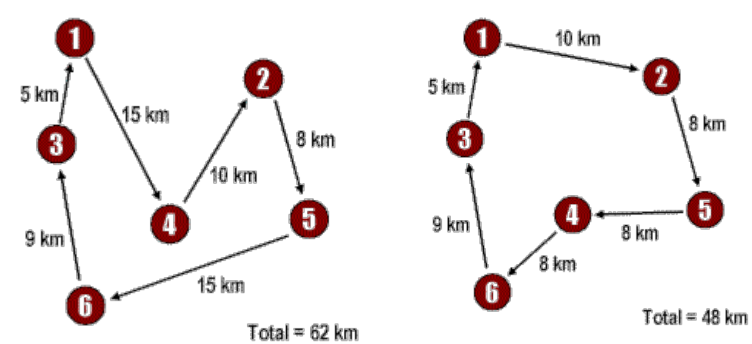
```
load_vector = 100:100:1000
section_h_vector  = 10:10:100
```

59. Giving the cities represented by letters A to F, and the distance among them represented by the value in the connecting line, calculate the shortest order to visit ALL the cities



Travel salesman problem solution:

- Acquire data from every city
- Calculate distance between all the cities (A-B, A-C, ... E-F)
- Try every possible combination
- Answer is the combination with the shortest sum



60. Sketch a problem of your own which you think that MATLAB can help to solve