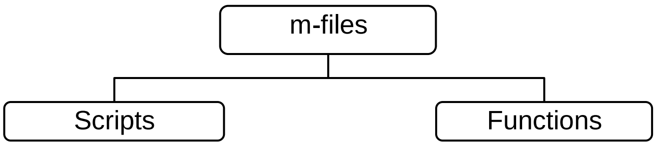
1. MATLAB m-files

Instead of typing commands directly in the command window, a series of commands may be placed into a file, and the entire file may be executed by typing its name in the command window. Such files are called as m-files. There are two kinds of m-files: script files and function files.



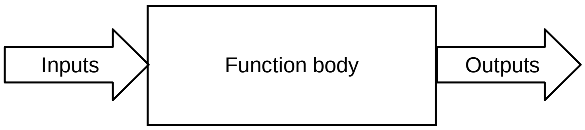
1.1 Script files

When you invoke a script, MATLAB simply executes the commands found in the file. Scripts can operate on existing data in the workspace, or they can create new data on which to operate. Although scripts do not return output arguments, any variables that they create remain in the workspace, to be used in subsequent computations. In addition, scripts can produce graphical output using functions like plot.

1.2 Function files

Functions are m-files that can accept input arguments and return output arguments. The names of the m-file and of the function should be the same. Functions operate on variables within their own workspace (local variables), separate from the workspace you access at the MATLAB command prompt. Function files are useful when you need to repeat a set of commands several times. The first line in a function file must begin with a function definition line that has a list of inputs and outputs. The function is called by typing its name at the command line followed by the input arguments between parentheses.

```
Function [output_variables]=function_name(input_variables)
```



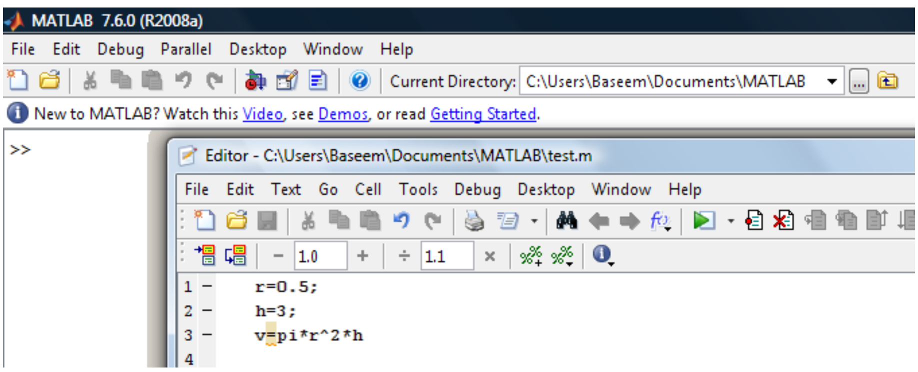
- To create new m-file, select File>New>m-file from the MATLAB file menu or click on the new m-file icon.
- To open an existing m-file, select File>Open from the MATLAB file menu or click on the open file icon.

Example 1: Write a script file to calculate the volume of a cylinder given its radius and height. Test your function with r=0.5 m and h=3 m.

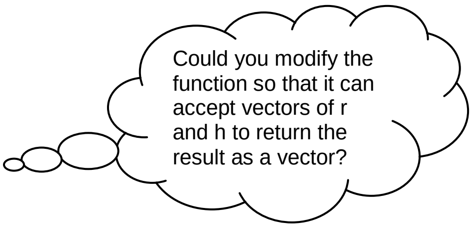
MATLAB session:

```
>> test
v =
2.3562
```

You must store your m-files in the MATLAB current directory.



Example 2: Write a function to calculate the volume of a cylinder given its radius and height. Test your function with r=0.5 m and h=3 m.

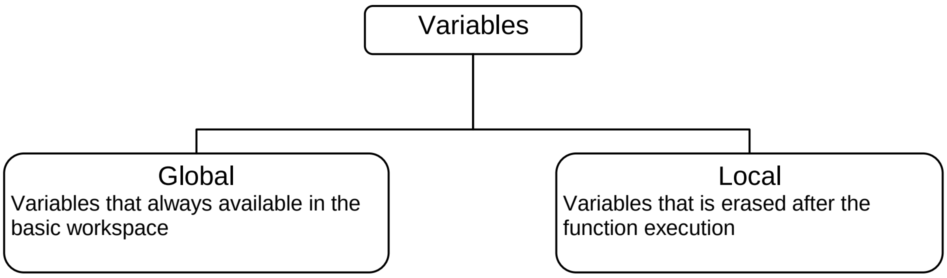


MATLAB session:

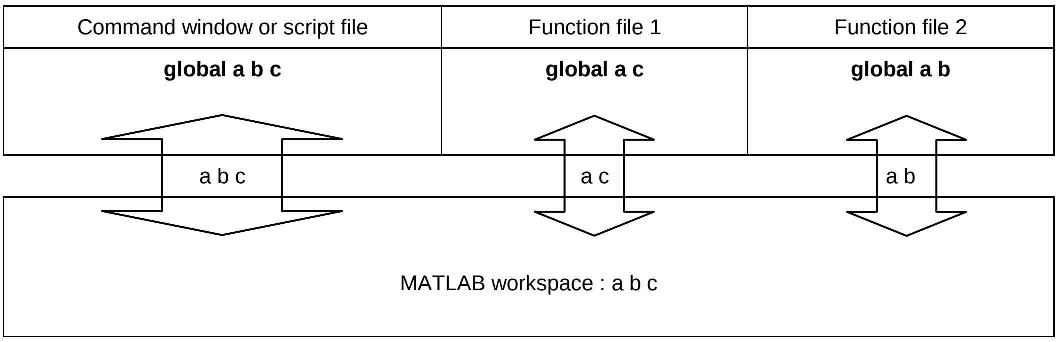
```
>> cyl_vol(0.5,3)
ans =
    2.3562
```

```
function v=cyl_vol(r,h)
% r   the cylinder radius
% h   the cylinder height
v=pi*r^2*h;
```

2. Global and local variables



Use the **global** command to share variables between the command window or script files and function files. The global command should be inserted at the beginning of each one to determine which variables are shared.



Example 3: Write a function that uses the ideal gas law to calculate the pressure of a gas at two temperatures if n=1, V=20 liter, at T=300 and 330 K.

MATLAB session:

```
>> global R
>> R=0.08206;
>> ideal(1,[300 330],20)
ans =
    1.2309    1.3540
```

```
function P=ideal(n,T,V)
global R
P=n*R*T./V;
```

3. Finding the zero of a function of one unknown

MATLAB uses the function fzero to solve the problem of f(x)=0 for x starting from an initial guess. The function syntax is:

```
fzero ('function_name',guess)
```

The name of the function file which contains the equation to be solved — `'function_name'`

An initial guess for the solution — `guess`

The solution steps are as follows:

1. Make the function equal to zero.
2. Define your function in a function file using the m-file editor (e.g. test.m).
3. Suggest an initial guess for the solution (e.g. 0).
4. Issue the following command in the command window or your script file: **`>>fzero('test',0)`**

   Then MATLAB will return a value of x that makes the function equal to zero.

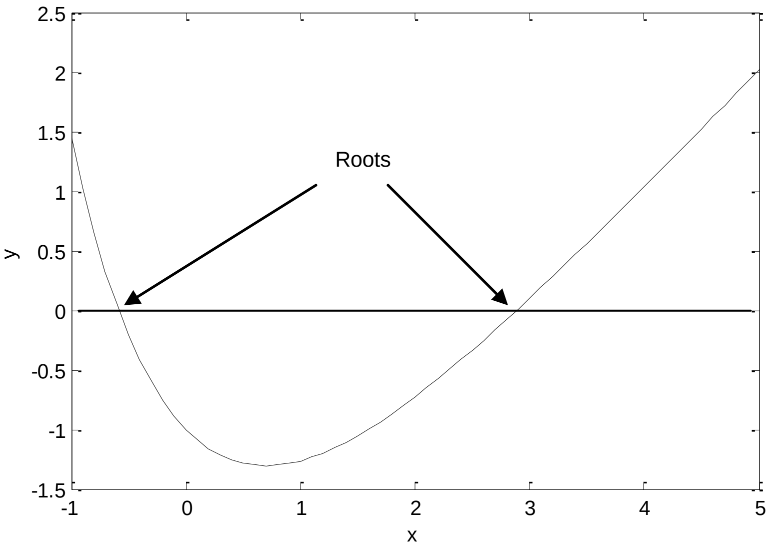
Example 4: Find the roots of the equation: $x + 2e^{-x} = 3$ near $x = -0.5$ and $x = 3$.

```
function f=ex4(x)
f=x+2*exp(-x)-3;
```

<u>MATLAB session:</u>

```
>> x=fzero('ex4',-0.5)
x =
   -0.5831
>> x=fzero('ex4',3)
x =
     2.8887
```

If the function have more than one solution, fzero will return the nearest solution to the guess



Example 5: Lee and Duffy (1976) relate the friction factor c_f for flow of suspension of fibrous particles to the Reynolds number Re by this empirical equation:

$$\sqrt{\frac{1}{c_f}} = \left(\frac{1}{k}\right)\ln\left(\mathrm{Re}\sqrt{c_f}\right) + \left(14 - \frac{5.6}{k}\right)$$

Comput c_f for Re=3750 and k=0.28.

<u>MATLAB session:</u>

```
>> global Re k
>> Re=3750;k=0.28;
>> guess=0.0396*Re^-0.25;
>> cf=fzero('ex5',guess)
cf =
     0.0051
```

```
function f=ex5(cf)
global Re k
f=sqrt(1/cf)-(1/k)*log(Re*sqrt(cf))-(14-5.6/k);
```

A good starting guess for the iterative solution of the equation may be found from the Blasius equation: $c_f = 0.0396\,\mathrm{Re}^{-0.25}$

4. Finding solution for a system of nonlinear equations

MATLAB uses the function fsolve to solve a system of nonlinear equations of several variables which takes the form f(x)=0 where f or x may be vectors or matrices. The function syntax is:

```
fsolve('function_name',guess)
```

function_name: The name of the function file which contains the equations to be solved

guess: An initial guess for each variable in the system.

Example 7: Solve the equations: (take 0.8 and 0.2 as initial guesses for x_1 and x_2)

$$x_1^2 + 4x_2^2 = 5$$

$$2x_1^2 - 2x_1 - 3x_2 = 2.5$$

MATLAB session:

```
function f = ex7(x)
f(1)=x(1)^2+4*x(2)^2-5;
f(2)=2*x(1)^2-2*x(1)-3*x(2)-2.5;
```

```
>> x=fsolve('ex7',[0.8 0.2])
Equation solved.
fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.
<stopping criteria details>
x =
    2.0000    0.5000
```

Example 8: For turbulent flow of fluids in an interconnected network, the flow rate v from one node to another is about proportional to the square root of the difference in pressures at the nodes. For the conduits shown, find the pressure at each node. The values of b represent conductance factors in the relation $v_{ij} = b_{ij}\sqrt{(p_i - p_j)}$.

Solution: setting up the equations for the pressures at each node gives:

Node 1:

$$0.3\sqrt{500 - p_1} = 0.2\sqrt{p_1 - p_2} + 0.1\sqrt{p_1 - p_3}$$

Node 2:

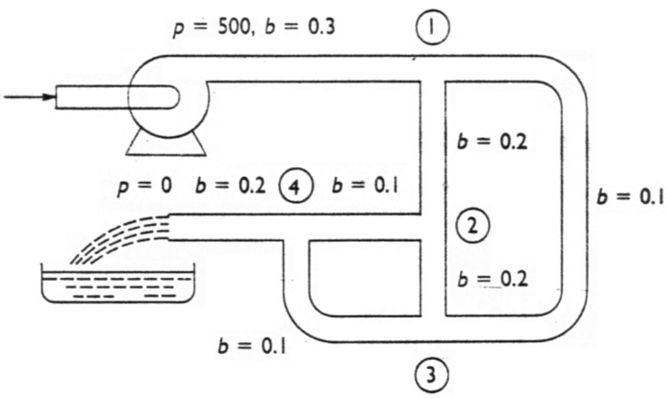
$$0.2\sqrt{p_1 - p_2} = 0.1\sqrt{p_2 - p_4} + 0.2\sqrt{p_2 - p_3}$$

Node 3:

$$0.1\sqrt{p_1 - p_3} + 0.2\sqrt{p_2 - p_3} = 0.1\sqrt{p_3 - p_4}$$

Node 4:

$$0.1\sqrt{p_2 - p_4} + 0.1\sqrt{p_3 - p_4} = 0.2\sqrt{p_4 - 0}$$



MATLAB session:

```
>> p=abs(fsolve('ex8',[400 200 200 0]))
Equation solved.
fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.
<stopping criteria details>
p =
  423.1898  347.7937  343.5118  172.8230
```

```
function f = ex8(p)
f(1)=0.3*sqrt(500-p(1))-0.2*sqrt(p(1)-p(2))-0.1*sqrt(p(1)-p(3));
f(2)=0.2*sqrt(p(1)-p(2))-0.1*sqrt(p(2)-p(4))-0.2*sqrt(p(2)-p(3));
f(3)=0.1*sqrt(p(1)-p(3))+0.2*sqrt(p(2)-p(3))-0.1*sqrt(p(3)-p(4));
f(4)=0.1*sqrt(p(2)-p(4))+0.1*sqrt(p(3)-p(4))-0.2*sqrt(p(4)-0);
```

PROBLEMS

1. Find a root of $f(x)=\sin x - x/2$, near $x=2.0$. (ans 1.8955)
2. DeSantis (1976)has derived a relationship for the compressibility factor of real gases of the form:

$$z=\frac{1+y+y^2-y^3}{(1-y)^3}$$

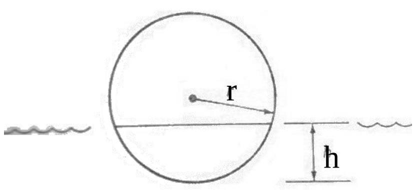
   where $y=b/(4v)$, b being the van der Waals correction and v is the molar volume. If $z=0.892$, what is the value of y? What answer is the realiable one if you take 0 and 1.5 as initial guesses.
3. In studies of solar-energy collection by focusing a field of plane mirrors on a central collector, Vant-Hull (1976) derives an equation for the geometrical concentration factor C:

$$C=\frac{\pi(h/\cos A)^2 F}{0.5\pi D^2(1+\sin A-0.5\cos A)}$$

   where A is the rim angle of the field, F is the fractional coverage of the field with mirrors, D is the diameter of the collector, and h is the height of the collector. Find A if h=300, C=1200, F=0.8, and D=14.
4. Based on the work of Frank-Kamenetski in 1955, temperatures in the interior of a material with embedded heat sources can be determined if we solve this equation:

$$e^{-(1/2)t}\cosh^{-1}(e^{(1/2)t})=\sqrt{\frac{1}{2}L_{er}}$$

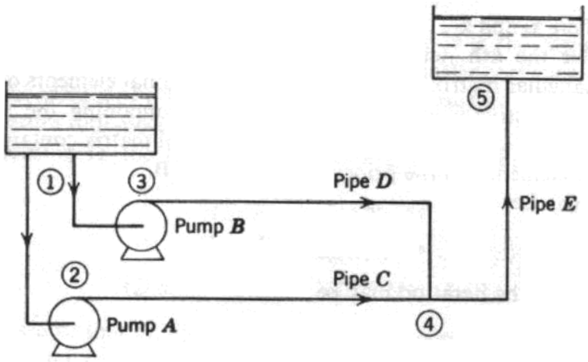
   Given that $L_{er}=0.088$, find t.
5. Using the formula for the volume of a segment of a sphere, it can be shown that the depth h to which a floating sphere of radius r sinks in water is a root of the equation: $h^3-3rh^2+4r^3d=0$. Where d is the specific gravity of the sphere. Suppose a wooden sphere if radius 0.5 m has specific gravity of 0.75. Calculate the depth (h) to which the sphere will sink.



6. Consider Q gpm of water flowing from point 1 to point 2 in an inclined pipe of length L ft and diameter D in. The equation below can be hold approximately for turbulent flow in pipes of average rouphness:

$$(z_2 - z_1) + 2.3(p_2 - p_1) + 8.69 \times 10^{-4} Q^2 L / D^5 = 0$$

Here, p is the pressure in psig and z is the elevation in feet. Also, a typical head-discharge curve for a centrifugal pump can be represented by:



$$\Delta p = \alpha - \beta Q^2$$

In which Δp is the pressure increase in psig across the pump, Q is the flow rate in gpm, and α and β are constants depending on the particular pump.

For the piping systemshown, the pressures p_1 and p_5 are both essentially atmospheric (0 psig); there is an increase in elevation between points 4 and 5, but pipes C and D are horizontal. Write a program that will accept values of $\alpha_A, \beta_A, \alpha_B, \beta_B$, $(z_5 - z_4)$, D_C, L_C, D_D, L_D, D_E, and L_E, and that will solve the resulting equations for the unknowns Q_C, Q_D, Q_E, p_2, p_3, and p_4. One suggested set of test data is $(z_5 - z_4)$ =70 ft, with the given data.

Pump	α, psi	β, psi/(gpm)²
A	156.6	0.00752
B	117.1	0.00427

Pipe	*D*, in.	*L*, ft
C	1.278	125
D	2.067	125
E	2.469	145

Assume that the above pipe lengths have already included the equivalent lengths of all fittings and valves.

7. When a pure sample gas is bombarded by low energy electrons in a mass spectrometer, the galvanometer shows peak heights that correspond to individual mass-to-charge ratios for the resulting mixture of ions. For the ith peak produced by a pure sample j,one can then assign a sensitivity s_{ij} (peak height per micron of Hg sample pressure). These coefficients are unique for each type of gas. A distibution of peak heights may also be obtained for n-components gas mixture that is to be analyzed for the partial pressures p_1, p_2, p_3, …,p_n on each of its constituents. The height of h_i of certain peak isa linear combination of the products of the individual sensitivities and partial pressures:

$$\sum_{j=1}^{n} s_{ij} p_j = 0$$

Write a program that will accept values for the number of components, the sensitivites, and peak heights. The program then should calculates the values for individual partial pressures from the following data. Take h_1=17.1, h_2=65.1, h_3=186, h_4=82.7, h_5=84.2, h_6=63.7, h_7=119.7. Check your answer with a total mixture pressure of 39.9 microns of Hg.

Peak Index	Component Index, *j*						
	1	2	3	4	5	6	7
i	Hydrogen	Methane	Ethylene	Ethane	Propylene	Propane	*n*-Pentane
1	16.87	0.1650	0.2019	0.3170	0.2340	0.1820	0.1100
2	0.0	27.70	0.8620	0.0620	0.0730	0.1310	0.1200
3	0.0	0.0	22.35	13.05	4.420	6.001	3.043
4	0.0	0.0	0.0	11.28	0.0	1.110	0.3710
5	0.0	0.0	0.0	0.0	9.850	1.684	2.108
6	0.0	0.0	0.0	0.0	0.2990	15.98	2.107
7	0.0	0.0	0.0	0.0	0.0	0.0	4.670