One-Dimensional Array: Vector

We have used scalar variables (a variable that stores one value at any time) up to this point. We now investigate the *array*, a collection of like values. An array has one name but multiple "bins" where each bin stores one value. For example, we may might be interested in the value of a variable x (such as the population of a city, the GDP of a nation, or the value of a stock portfolio) at different points in time over some interval. Quite likely we'll call these values x_1, x_2, and x_3, etc. – there is one variable name and we identify the individual values with the subscripts 1, 2, and 3. The corresponding data structure in programming is the *one-dimensional array*.

In MATLAB, one dimensional arrays are called *vectors*. The "subscript" that identifies the "bin number" is called the *index* or cell position. MATLAB **array index starts at 1**, not zero. To access a value in an array, use parentheses to enclose the index value. For example, x(2) is the value in the 2nd cell of vector x. Use *square brackets* to delimit vectors.

MATLAB also distinguishes between *row* and *column* vectors. In a *row* vector, the bins are arranged from left to right; in a *column* vector they are stacked from top to bottom. Usually, row and column vectors are interchangeable: however, sometimes the type of vector you are using is important. Fortunately, Matlab has a handy *transpose* operation, denoted by ', which allows you to change a row vector into a column vector, or vice versa.

Type the following statement in the Command Window. Observe the results and fill in the blanks.

```
x= [11 23 9]    % Row or column? _____
y= [11; 23; 9]  % Row or column? _____
y= y'           % What does the symbol ' do? _____
y(3)            % Access the value in cell 3 of vector y.
y(5)            % What is the error? _____
y(5)= 8         % ASSIGNMENT is different from ACCESS (previous expression).
                %   This works because you create variables simply by
                %   assigning values to it.  What is in y(4)?  _____
length(y)       % _____
[m,n]= size(y)  % m is _____; n is _____
```

Write down the vectors that get created by the statements on the left below.

```
% Create vectors using functions or short-cut expressions
  a= zeros(1,5)      % 1 row, 4 columns (row vector) of zeroes
  b= ones(5,1)       % 5 rows, 1 column (column vector) of ones
  c= rand(1,4)       % 1 row, 4 columns of random numbers between 0 and 1
  d= 1:4
  e= 1:3:11
  f= linspace(0,1,5)

% Build vectors by concatenation
  x= ones(1,3)
  y= [2 4]
  z= [x y]
  z= [z 0]
  z= [9 z z]
  c= [zeros(1,2)'; 6]

% Sub-vectors
  x= [2 5 8 6 9]
  x(1:3)
  x(3:length(x))
  x(3:end)
  x([1,3,4])
```