

Matlab Notes

Let's start with some simple stuff:

Define variables:

```
>> a=5
```

```
a =
```

```
5
```

```
>> b=6
```

```
b =
```

```
6
```

Now let's do something with them:

```
>> a+b
```

```
ans =
```

```
11
```

```
>> a/b
```

```
ans =
```

```
0.8333
```

```
>> b^2
```

```
ans =
```

```
36
```

```
>> a^0.5
```

```
ans =
```

```
2.2361
```

```
> > sin(a)
```

```
ans =
```

```
-0.9589
```

What a strange answer? Why? Let's look at the sin of a strange number:

```
>> c=2*pi
```

```
c =
```

```
6.2832
```

```
>> sin(c)
```

```
ans =
```

```
-2.4493e-16
```

The first important aspect of this example is to note that Matlab measures angles in radians ($360^\circ = 2\pi$ radians), as do most (if not all) software languages. The second aspect to note is that the answer is not 0, as it should be, however it is very close. This highlights a problem with numerical methods for the solution of problems – they may not be exact, but can be very close to exact. We must be aware of this limitation and take it into account when we use Matlab (more on this later).

The real strength of Matlab is its ability to perform linear algebra. Let us now create matrices and use them to solve problems.

First create a matrix (note use of “,” “;” and “:” in next lines)

```
A = [2,0;1,1]
```

```
2. 0.
```

```
1. 1.
```

Now extract row 2:

```
-->r=A(2,:)
```

```
r =
```

```
1. 1.
```

How to extract a column from a matrix in scilab:

Let's extract the first column:

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

```
R =
```

```
2.
```

```
1.
```

Now for some more complex tasks that by hand might drive your crazy

Example of how to solve simultaneous algebraic equations:

$$3x + 4y + 2z = 44$$

$$7x + 2y + 8z = 82$$

$$x + 3y + 2z = 34$$

```
-->a=[3,4,2;7,2,8;1,3,2]
```

```
a =
```

```
3.  4.  2.  
7.  2.  8.  
1.  3.  2.
```

```
-->b=[44;82;34]
```

```
b =
```

```
44.  
82.  
34.
```

```
-->a\b
```

```
ans =
```

```
2.  
6.  
7.
```

Example of root finding:

$$6x^2 + 17x + 6 = 64$$

$$6x^2 + 17x - 58 = 0$$

```
-->y=roots([6,17,-58])
```

```
y =
```

```
2.  
- 4.8333333
```

```
-
```

Testing the solution – something you must always do!

```
-->y1=y(2)
```

```
y1 =
```

```
- 4.8333333
```

```
-->6*y1^2+17*y1-58
```

```
ans =
```

```
- 2.842E-14
```

Finding the roots of another polynomial:

$$3x^3+6x^2 + 17x + 6 = 64$$

$$3x^3 + 6x^2 + 17x - 58 = 0$$

```
-->y=roots([3,6,17,-58])
```

```
y =
```

```
1.6522596
```

```
1.8261298 + 2.8924726i
```

```
-1.8261298 - 2.8924726i
```

Note that the roots come out as imaginary numbers and we are typically only interested in the real numbers

The real power in computers is their ability to do the same thing time after time after time. These are often called “loops” so let us look at how to construct and use them with some simple examples:

Loops in Matlab

```
x=1;for k=1:4,x=x*k,end
```

```
x =
```

```
1.
```

```
x =
```

```
2.
```

```
x =
```

```
6.
```

```
x =
```

```
24.
```

Now write a program to save that has a while loop:

```
x=0;
```

```
while x < 100
```

```
sum = x + 1;
```

```
x=x+1;
```

```
end
```

```
fprintf('%d\n',sum);
```

Matlab can also make some beautiful plots

Now write a program that does a simple plot (`//` starts a comment line):

```
%% x initialisation
x=[0:0.1:2*%pi]';
%%simple plot
%% plots index number in the x and sin(x) in the y
plot(sin(x))
```

```
x=[0:0.1:2*%pi]';
plot(x,sin(x));
```

another example:

```
-->lg=[372.15,-227.906;373.15,-225.106;374.15,-224.306]
```

```
lg =
```

```
372.15 - 227.906
```

```
373.15 - 225.106
```

```
374.15 - 224.306
```

```
plot(lg(:,1),lg(:,2))
```