

Octave -An Open-Source tool, similar to Matlab

KR Chowdhary
Former Professor & Head
Email: kr.chowdhary@gmail.com
Webpage:www.krchowdhary.com

Department of Computer Science and Engineering
MBM Engineering College, Jodhpur

May 27, 2015

- ▶ Open source interactive software tool for numerical computations and graphics
- ▶ Designed for matrix multiplication
- ▶ Solving Simultaneous equations, computing Eigen-vectors and Eigen-values
- ▶ Data can be expressed in matrix and vectors
- ▶ data can be displayed in various formats, has its own programming language
- ▶ More suited for Engineering problem solution, and equivalent to Matlab but open source
- ▶ Far more efficient than C++ or other HLLS
- ▶ Octave has command-line interface
- ▶ Octave is interpreted language

```
$ octave <cr>  
octave:1> 2+2  
ans=4
```

```
octave:1> exp(1)  
ans=2.7183
```

```
octave:##> 1.2*sin(40*pi/180+log(2.4^2))  
ans=0.76618
```

```
octave:##> who  
ans
```

```
octave:##> format long  
octave:##> ans  
ans=0.766177651029692
```

```
octave:##> format short  
octave:##> ans  
=0.76618
```

cos: Cosine of an angle (in radians)

sin: Sine of an angle (in radians)

tan: Tangent of an angle (in radians)

exp: Exponential function (e^x)

log: Natural logarithm (NB this is \log_e , not \log_{10})

log10: Logarithm to base 10

sinh: Hyperbolic sine

cosh: Hyperbolic cosine

tanh: Hyperbolic tangent

acos: Inverse cosine

acosh: Inverse hyperbolic cosine

asin: Inverse sine

asinh: Inverse hyperbolic sine

atan: Inverse tangent

atan2: Two-argument form of inverse tangent

atanh: Inverse hyperbolic tangent

abs: Absolute value

sign: Sign of the number (-1 or +1)

round: Round to the nearest integer

floor: Round down (towards minus infinity)

ceil: Round up (towards plus infinity)

fix: Round towards zero

rem: Remainder after integer division

Loading and saving data files:

```
octave:1> save anyname
```

(saves to anyname.mat, in current directory)

Later on it can be retrieved by:

```
octave:1> load anyname
```

```
octave:##> help sqrt
```

```
octave:##> a=[1 4 5]
```

```
a =
```

```
  1   4   5
```

```
octave:##> b=[2, 1, 0]
```

```
b= 2  1  0
```

```
octave:##> c= [4; 7; 10]           is col. vecor
```

```
c = 4
```

```
    5
```

```
   10
```

```
octave:##> e = 2:6
```

```
e=2 3 4 5 6
```

```
octave:##> e=2:0.3:4
```

```
e=2.000  2.300  ....
```

```
3.800
```

```
octave:##> a=[1:2:6 -1 0]
a=1 3 5 -1 0
```

```
octave:##>a(3)
ans= 5
```

```
octave:##> a(3:5)
5 -1 0
```

```
octave:##> a*2
2 6 10 -2 0
```

```
octave:##> b=[1 2 3 4 5 6]
b= 1 2 3 4 5 6
```

```
octave:##> a.^2
ans=1 9 25 1 0
```

```
octave:##> b.^2
ans=1 4 9 16 25 36
```



```
octave:##> angles=[0:pi/3:2*pi]
angles=0.0000    ....        6.28319
```

```
octave:##> y=sin(angles)
y=0.0000    ....    0.86603    .....-0.0000
```

Plotting Graph by command:

```
octave:##> plot(angles, y)
see that graph is displayed. It can be saved
from display.
```