Basics

Variables and Expressions
Assignment Statements
Built-In Functions
Scripts
Comments
Keyboard Input
Formatting Output

Approach

Preview key concepts by first playing with Matlab as a calculator.

From formula to program.

Three Formulas

- Surface area of a sphere? $A = 4\pi r^2$
- Have the cosine of some angle $\theta \in [0, \pi/2]$ and want $\cos(\theta/2)$? $\cos(\theta/2) = \sqrt{\frac{1+\cos(\theta)}{2}}$
- Need the roots of a quadratic function? $r = \frac{-b \pm \sqrt{b^2 + 4ac}}{2a}$

Surface Area Increase

In the Command Window…

```
>> r = 6365;
>> delta = .000001;
>> A_plus = 4*pi*(r+delta)^2;
>> A = 4*pi*r^2;
>> Increase = A_plus - A
Increase =
        0.159969925880 43
```

Cosine(15 degrees)

```
>> c = cos(pi/3);
>> c = sqrt((1+c)/2);
>> c = sqrt((1+c)/2)
c =
    0.96592582628907
>> c15 = cos(pi/12)
c15 =
    0.96592582628907
```

$X^2 + 5x + 6 = (x+2)(x+3)$

```
>> a = 1;
>> b = 5;
>> c = 6;
>> d = sqrt(b^2 - 4*a*c);
>> r1 = (-b - d)/(2*a)
r1 =
     -3
>> r2 = (-b + d)/(2*a)
r2 =
     -2
```

Let's revisit the key ideas above and introduce others…

A Script

```
% Quad1
% Solves  x^2 + 5x + 6 = 0


a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Script

A sequence of instructions.

The order of the instructions is important.

A script is a program.

Comments

```
% Quad1
% Solves  x^2 + 5x + 6 = 0

a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Comments

Begin with the “%” symbol. Goes to the end of the line.

Facilitate the reading and understanding of the script.

Comments and Readability

Start each program (script) with a concise description of what it does

Define each important variable/constant

Top a block of code for a specific task with a concise comment.

Arithmetic Expressions

```
% Quad1
% Solves  x^2 + 5x + 6 = 0

a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Insight Through

Arithmetic Expression

A recipe that results in the production of a number.

Built-In Functions

```
% Quad1
% Solves  x^2 + 5x + 6 = 0

a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Built-In Functions

These are "packagings" of more advanced calculations.

Some examples: log, exp, sin, cos,…

Variables

```
% Quad1
% Solves  x^2 + 5x + 6 = 0

a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Variables

- A variable is a "box" holding a numerical value.
- It has a name.
- Names must begin with a letter.
- Names are case sensitive.
- Names can combine letters, numbers, underscore.

  Example: **`x1A_New`**

Assignment Statements

```
% Quad1
% Solves  x^2 + 5x + 6 = 0


a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

Assignment Statements

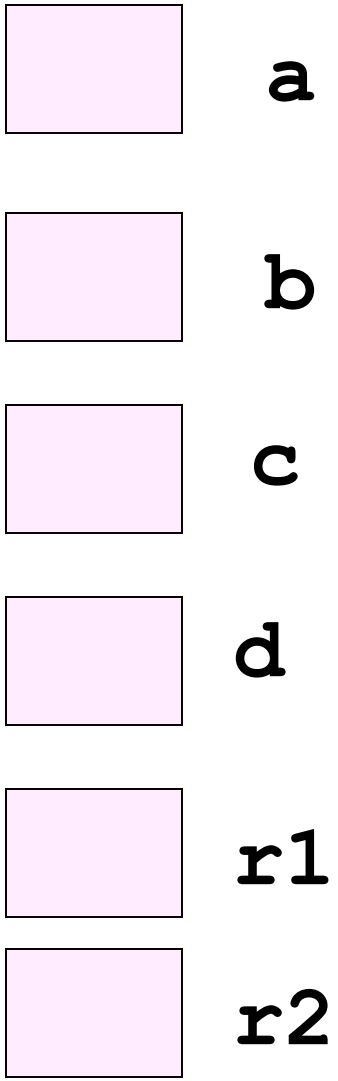
Variable Name = Arithmetic Expression

where to put the value

a recipe to compute the value

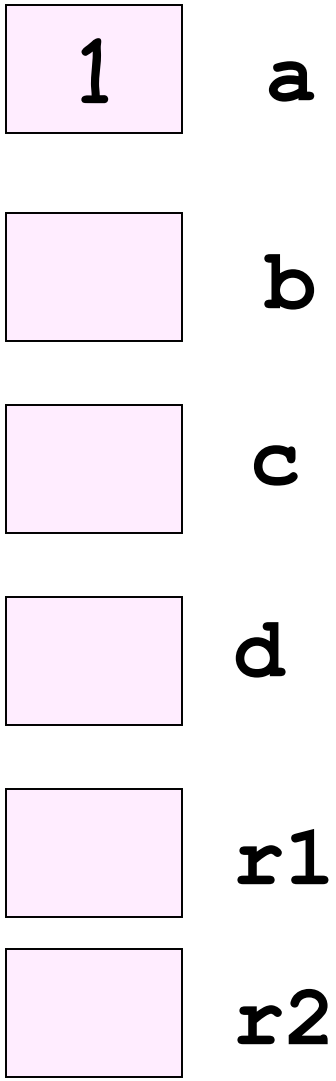
Script Execution

```
a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```



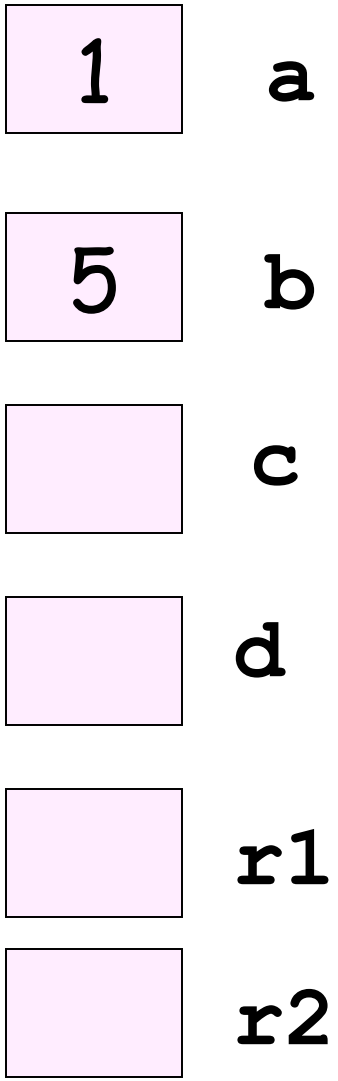
Script Execution

```
a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```



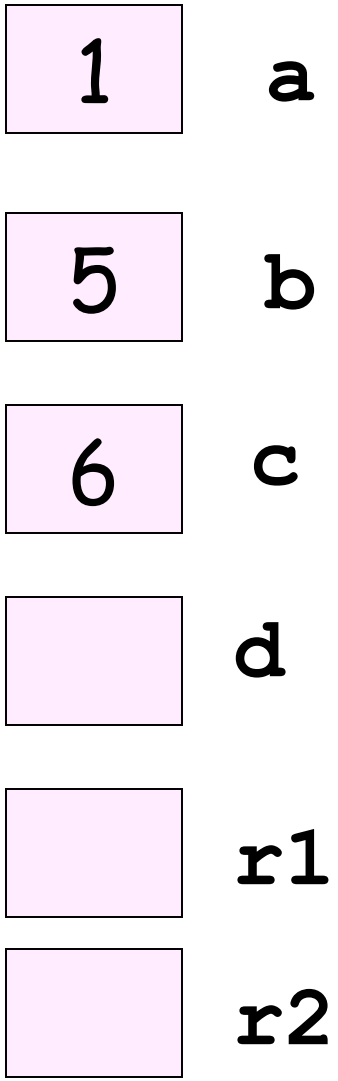
Script Execution

```
a   = 1;
b   = 5;
c   = 6;
d   = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```



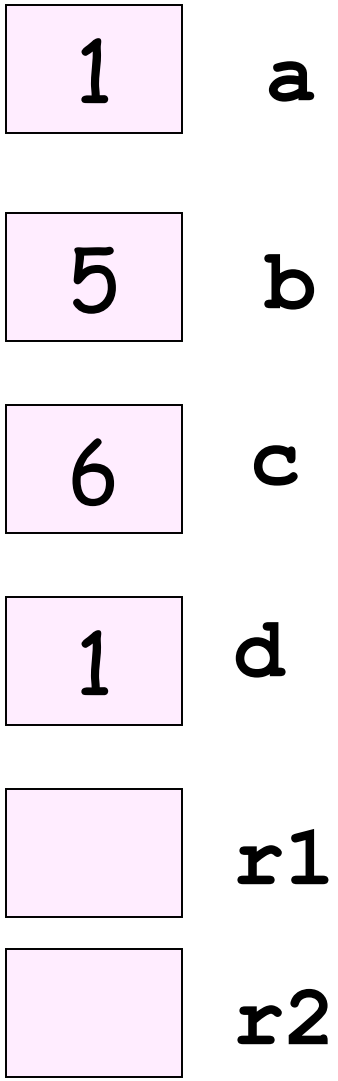
Script Execution

```
a  = 1;
b  = 5;
c  = 6;
d  = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```



Script Execution

```
a   = 1;
b   = 5;
c   = 6;
d   = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```



Script Execution

```
a   = 1;
b   = 5;
c   = 6;
d   = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

1	a
5	b
6	c
1	d
-3	r1
	r2

Script Execution

```
a   = 1;
b   = 5;
c   = 6;
d   = sqrt(b^2 -4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

1	a
5	b
6	c
1	d
-3	r1
-2	r2

Remember…

Instructions are executed in order.

The right hand side is evaluated first;

That value is assigned to the variable named on the left hand side.

Variables on the right hand side must have values before being used.

Question Time

What is the value of **X** and **Y** after the following script is executed:

```
X = 2;
Y = 7*X;
X = Y;
X = X + 1;
```

A: X is 5 and Y is 14

C: X is 5 and Y is 21

B: X is 15 and Y is 14

D: X is 15 and Y is 2

Question Time

What is the final value of **X** and **Y** ?

```
> X = 8;
> Y = X;
> X = Y;
> X = 2*X;
> Y = Y/2;
```

A: X is 16 and Y is 16

C: X is 16 and Y is 4

B: X is 8 and Y is 8

D: X is 8 and Y is 4

Another Script

```
% Quad2
% Solves ax^2 + bx + c = 0
% Assumes real roots.

a = input('Enter a: ');
b = input('Enter b: ');
c = input('Enter c: ');
d = sqrt(b^2 - 4*a*c);
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

The `input` Command

Variable Name = input(' Message');

where to put the value

a prompt message in quotes

Processed after the user hits the <enter> key.

Formatting Output

When leaving off the semicolon isn't good enough.

The tools: **`disp`**, **`fprintf`**

disp

Displays a string.

Example:

```
disp('This is a message')
```

fprintf

Used to format output. Example:

```
x = 1.23456789;
fprintf('x = %5.2f\n',x)
```

Output line will look like

```
x =  1.23
```

The \n generates a carriage return

A Modification…

```
r1 = (-b - d)/(2*a)
r2 = (-b + d)/(2*a)
```

```
r1 = (-b - d)/(2*a);
r2 = (-b + d)/(2*a);

disp(' ')
fprintf('Root1 = %10.6f\n',r1))
fprintf('Root2 = %10.6f',r2))
```