- Previous Lecture (and Lab):
  - Intro to the course, "Computational senses"
  - The Matlab Command Window
- Today's Lecture:
  - Anatomy of a program
  - Variables, assignment, mathematical operations
  - Functions for input & output
- Announcements
  - Due to the fixed lab capacity, you must attend the discussion section in which you are enrolled
  - Consulting begins Tuesday in ACCEL Green Room (Carpenter Hall)

Formula

- Surface area of a sphere?

Formula

- Surface area of a sphere?

$$A = 4\pi r^2$$

Formula

- Surface area of a sphere? $$A = 4\pi r^2$$
- Have the cosine of some angle and want $\cos(\theta/2)$? $$\theta \in \left[0, \pi/2\right]$$

Formula

- Surface area of a sphere?

$$A = 4\pi r^2$$

- Have the cosine of some angle and want $\cos(\theta/2)$?

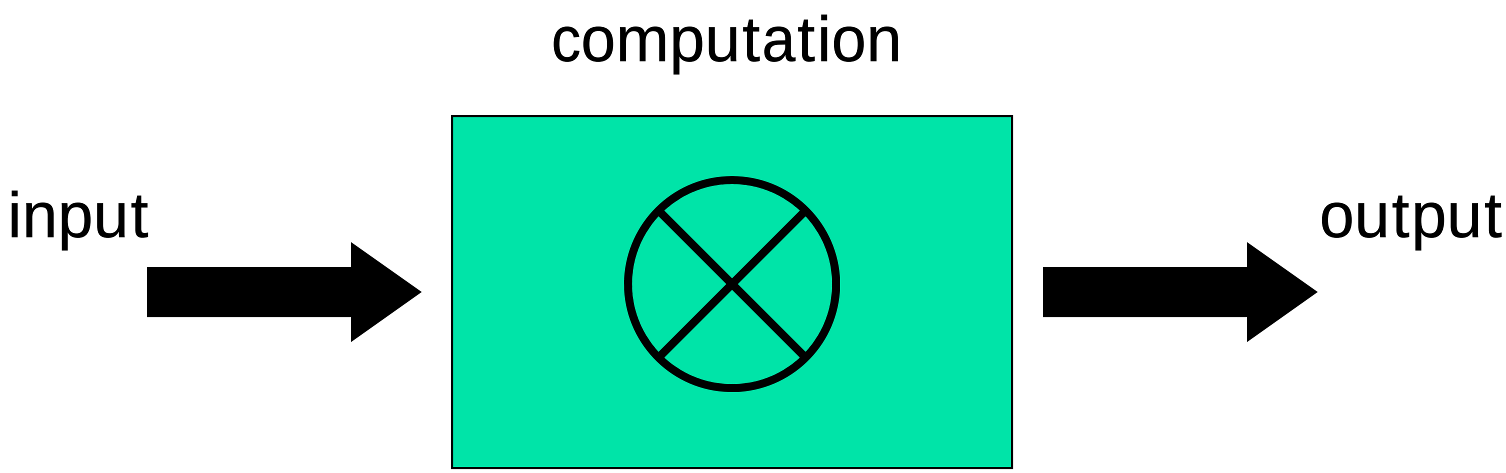
$$\theta \in \left[0, \pi/2\right]$$

$$\cos(\theta / 2) = \sqrt{\frac{1 + \cos(\theta)}{2}}$$

```
% Example 1_1:  Surface area of a sphere
% A: surface area of the sphere
% r: radius of the sphere

r= input('Enter the radius: ');
A= 4*3.14159*r*r;
fprintf('Surface area is %f!\n', A)
```

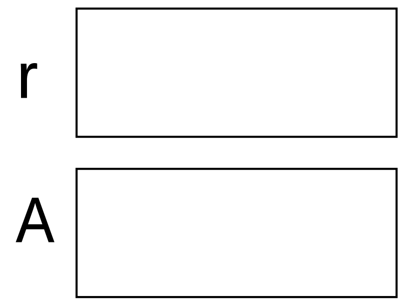
```
% Example 1_1:  Surface area of a sphere
% A: surface area of the sphere
% r: radius of the sphere

r= input('Enter the radius: ');
A= 4*pi*r*r;
fprintf('Surface area is %f!\n', A)
```

A computer program



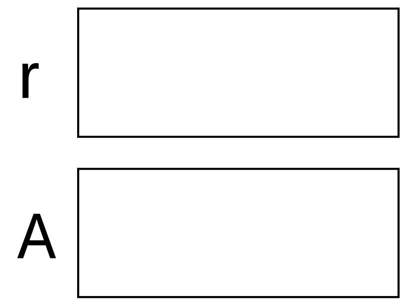
Variable & assignment

- Variable: a named computer memory space for storing a value



Variable & assignment

- Variable: a named computer memory space for storing a value



- Valid names start with a letter, can contain digits
- Use meaningful variable names!

Variable & assignment

- Variable: a named space for storing a value

r 

- Assignment: putting a value into a variable
- Assignment operator: `=`
- An assignment statement: `r= 2*4.5`
- Expression on right-hand-side (rhs) is evaluated before the assignment operation

Assignment

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 – cos(y)
```

Assignment

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 – cos(y)
```

- Question: can we reverse the order of the 3 statements above?

Assignment

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 – cos(y)
```

- Question: can we reverse the order of the 3 statements above?
- NO! Any variable on the rhs must be initialized.

Assignment

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 – cos(y)
```

- Question: can we reverse the order of the 3 statements above?
- NO! Any variable on the rhs must be initialized.

Matlab's built-in functions

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 - cos(y)
```

- Question: can we reverse the order of the 3 statements above?
- NO! Any variable on the rhs must be initialized.

Matlab's built-in functions

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 - cos(y)
```

Function name

Argument (value) passed to the function

- Question: can we reverse the order of the 3 statements above?
- NO! Any variable on the rhs must be initialized.

Matlab's built-in functions

- Expression on rhs is evaluated before the assignment operation
- Examples:

```
x= 2*3.14
y= 1+x
z= 4^2 - cos(y)
```

- Question: can we reverse the order of the 3 statements above?
- NO! Any variable on the rhs must be initialized.

Script execution
(A script is a sequence of statements, an “m-file”)

```
% 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)
```

Memory space

Script execution

(A script is a sequence of statements, an "m-file")

```
% 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)
```

Memory space

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

Statements in a program are executed in sequence

```
% A program fragment ...
x= 2*3.14
y= 1+x
x= 5
% What is y now?
```

A: 6 | B: 7.28 | C: *some other value, or error*

```
% Example 1_1:  Surface area of a sphere
% A: surface area of the sphere
% r: radius of the sphere

r= input('Enter the radius: ');
A= 4*3.14159*r*r;
fprintf('Surface area is %f!\n', A)
```

Input & output

- *variable* = **input**('*prompt* ')

- **fprintf**('*message to print* ')

Input & output

- *variable* = **input**(‘*prompt* ’)

```
r= input(‘Enter radius: ’)
```

- **fprintf**(‘*message to print* ’)

```
fprintf(‘Increase ’)
fprintf(‘is %f inches\n’, x)
fprintf(‘Position (%d,%d)\n’, x,y)
```

Substitution sequences (conversion specifications)

`%f`	fixed point (or floating point)
`%d`	decimal—whole number
`%e`	exponential
`%g`	general—Matlab chooses a format
`%c`	character
`%s`	string

Examples: **`%f`** **`%15.2f`**

Comments

- For readability!
- A comment starts with % and goes to the end of the line
- 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

Example

Modify the previous program to calculate the increase in surface area given an increase in the radius of a sphere.

Note: 1 mile = 5280 feet

```
% Example 1_2:  Surface area increase
% given an increase in the radius

r= input('Enter radius r in miles: ');
delta= input('Enter delta r in inches: ');
```

```
% Example 1_2:  Surface area increase
% given an increase in the radius

r= input('Enter radius r in miles: ');
delta= input('Enter delta r in inches: ');
newr= r + ((delta/12)/5280);
A= 4*pi*r^2;
newA= 4*pi*newr^2;
incr= newA - A;
fprintf('Increase in mile^2 is %f.\n', incr)
```