3 Conditionals

All programming languages have conditionals.

3.1 If

Logical expressions are either true (equal to 1) or false (equal to 0). The simplest form of a conditional statement uses an if to evaluate a logical expression and then execute some code only if the logical expression evaluated to true. For example:

```matlab
if ( numItems == 0 )
    disp( 'Cannot compute average' );
end
```

Note the double-equals (which cannot be separated by a space)! When these lines of code are executed, if numItems is zero, we will get the error message; otherwise there will be no message. The end indicates where the “body” of the conditional stops: there could be multiple statements that are only executed if the condition is true. The brackets around the condition are optional.

One can also provide code to be executed when the condition is false. Here is code to calculate the absolute value of num (though of course there is a built-in command):

```matlab
if num > 0
    abs = num;
else
    abs = -num;
end
```

And then one can even do a three-way conditional using:

```matlab
if ... elseif ... else ... end
```

3.2 Relational Operators

Logical variables are also known as boolean variables. One can combine logical variables and expressions using relational operators. MATLAB uses ~ for not, & &
for and, and || for (inclusive) or. (Technically there is also a single & and a single |, but the double-version is recommended because it does short-circuit calculation: not evaluating the second part unless necessary.)

---

**Sample code**

*Here is code to report where a value is within range.*

```matlab
% script: isInRange
% goddard 2020
minVal=2;
maxVal=22;
fprintf("min is %d max is %d\n", minVal, maxVal);
x = input('give me a value ');
if (x >= minVal) && (x <= maxVal)
    disp('Value within specified range. ');
elseif (x > maxVal)
    disp('Value exceeds maximum value.');
else
    disp('Value is below minimum value.');
end
```