

**IF SELECTION**

Tommy Luckner  
Department of Mathematics

**OVERVIEW**

The goal of this week's lab is to learn about the `if` control structure and use it to add error checks to some of the previously developed functions.

**ACTIVITY**if selection:

- If there are only two cases to consider, then the general form of the `if` statement is:

```
if ( condition )
    ...MATLAB commands # 1
else
    ...MATLAB commands # 2
end
```

If “condition” is true, “MATLAB commands # 1” will be executed; if “condition” is false, “MATLAB commands # 2” will be executed. If you want MATLAB to do nothing if “condition” is false, then you can omit the “else” portion.

- If there are three or more cases to consider, then the general form of the `if` statement is:

```
if ( condition # 1)
    ...MATLAB commands # 1
elseif ( condition # 2 )
    ...MATLAB commands # 2
else ( condition # 3 )
    ...MATLAB commands # 3
end
```

For the case of three or more cases you often end with an “else” instead of an “elseif” statement. Examples of conditions:

```
a < b      a > b      a == b      a <= b      a >= b      a ~= b
          (a <= b && a ~= b)      (a < b || a == b)
```

Example if statement:

```
>> x = 2;
>> y = 3;
>> if y < x
    disp('x is greater than y. ');
else
    error('x is less than y. ');
end
```

### IN-CLASS EXERCISE

Modify `mydot.m` to check if the inputs are vectors and if their dimensions match by only using the dimensions of the inputs. Test your function with the following vector and matrix inputs.

$$u = (3 \ 4), \quad v = (1 \ 2 \ 3), \quad w = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

$$x = (1 \ 2), \quad y = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

---