Extra/Advanced MATLAB Notes

- statistics, data, and fitting

```
>> p = [3  0  -1];
>> polyval(p, 1:2)
  ans =
          2   11
```

$3x^2 + 0x + -1 \Rightarrow 3(1)^2 + 0(1) + -1 = 2$

$\Rightarrow 3(2)^2 + 0(2) + -1 = 11$

```
>> set1 = 2:5;
>> set2 = [4  0  -3  2];
>> intersect(set1, abs(set2))
  ans =
          2  3  4
```

set1 ⇒ 2 3 4 5

abs(set2) ⇒ 4 0 3 2

(since no 'stable' passed, automatically sorts)

```
>> ismember(set1, set2)
  ans =
          1  0  1  0
```

logical array

```
>> vec1 = 3:6;
>> median(vec1)
  ans =
          4.5
```

3 4 ↑ 5 6

4.5

```
>> mode([vec1  2:4])
  ans =
           3
```

3 4 5 6 2 3 4

* always the smallest mode if there's multiple *

- anonymous functions
  - → useful for quick calculations, such as conversions
  - → don't need separate code files!

```
functionhandle = @(x)  x.^2 + 5.*x + 5;
example_call = functionhandle(1)
     example_call =
                  11
```

ex.

```
anony = @(x, a)  4 * x - a;
anony(5, 2)
              18

anony(1:4, 3)

              1   5   9   13
```

$4 \cdot 1 - 3 = 1$

$4 \cdot 2 - 3 = 5$

$4 \cdot 3 - 3 = 9$

$4 \cdot 4 - 3 = 13$

- text manipulation

strtok:

strtok(strvar, delimiter)
note that default delimiter is a space

ex.

```
>> examplestring = 'Well hello there!';
>> [f rest] = strtok(examplestring)
    f =
        Well
    rest =
        _hello_there   %includes delimiter (space)
```

strcat:

strcat(str1, str2)
concatenates str2 to end of str1

ex.

```
>> strcat('Well_', 'hello!')
    ans =
        'Wellhello'   % removes trailing blanks
                        for character vectors

>> strcat("Well_", "hello!")
    ans =
        "Well_hello!"  %. keeps trailing blanks
                           for strings
```

strrep:

strrep(string, oldstring, newstring)
* replaces all oldstring occurences with newstring and the length of newstring ~= oldstring *
* works for strings and character vectors *

ex.

```
>> strrep('hello', 'lo', 'p!')
    ans =
        help!
```

sprintf:

just like fprintf, can be saved into a variable
* useful for titling (plots, labels, etc.) and for user input prompts *

Functions that work with strings but not character vectors:

+, strings(), strjoin(), strsplit(), join()

- text manipulation tricks

```
>> sports = ["baseball" "football" "track"];

>> "I love_" + sports
  ans =
        ["I love baseball" "I love football" "I love track"]

>> sports{2}(1:3)
  ans =
        'foo'

>> sports(3)
  ans =
        "track"
```