Functions in Python

Shankar Kulumani

Flight Dynamics & Control Lab

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

What is a function?

- Series of evaluations with defined inputs adn outputs
- Allows for code to be organized
- Deterministic - fixed input gives a fixed output
- Easier to test
- Easier to manipulate and use - efficient building block for more complex programs

Built-in Functions

- Before using a funciton you must either define it or import it into Python
- Read the documentation of a function to understand what to use it - inside Ipython

```
import numpy
numpy.sin?
```

- Calling a function is easy - `out = function(in)`
- Many already built into Python - others inside packages (Numpy, Scipy, your own, etc.)

How do you write a function?

- Functions can be written in any text editor
- Only need to use spaces (instead of Tabs)
- Best to have a “good” editor
  - Automatic indentation - very important in Python
  - Syntax highlighting - pretty colors
  - Autocompletion - for the lazy

Example Function

```
def function_name(input_1, input_2):
    """Documentation

    """

    output_1 = input_1
    output_2 = input_2

    return output_1, output_2

output_1, output_2 = function_name(input_1, input_2)
```

Function scope

- Variable available inside of a function
  - Inputs passed into functions
  - Module level variables

Script vs. Function

- Function requires inputs - must be called properly
- Scripts - can be run interactively
- Variables available to interactive interpreter

Example Function

- Write a function that converts fahrenheight to celsius
- Write a function that converts celsius to fahrenheit
- Write unit tests for both

$$F = 1.8C + 32 \tag{1}$$

$$C = \frac{F - 32}{1.8} \tag{2}$$