Introduction to Programming

Nick Marchuk
Senior Lecturer, NU Mechanical Engineering
Ford B100
nick.marchuk@u.northwestern.edu

While I’m talking

- Find a copy of this document at hades.mech.northwestern.edu
- Download and install Mu from codewith.mu
- Download the CSV file of the CTA Ridership file from https://data.cityofchicago.org/Transportation/CTA-Ridership-L-Station-Entries-Daily-Totals

Programming Languages

Programming Languages

- Compiled Languages
  - C / C++
  - For a specific architecture, runs fast, slowest to write
- Statically Interpreted Languages
  - Java
  - Cross-platform, runs slow, faster to write
- Dynamically Interpreted Languages
  - Python, MATLAB
  - Runs slowest, fastest to prototype

Professional Programming Development

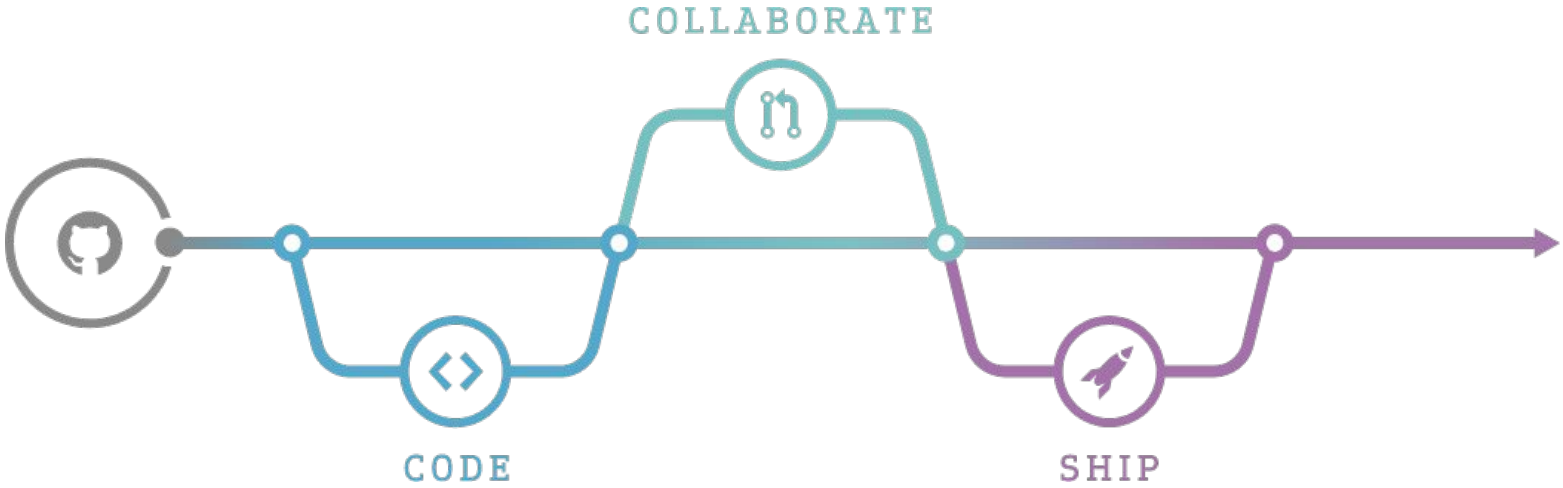
- On Windows: Visual Studio
- On OSX: Xcode
- On either
  - Netbeans, Eclipse
- For iOS: Swift using Xcode (like C++)
- For Android: Android Studio (based on Java)

Programming for Prototypes

- Goal: test an algorithm, automate a task, filter data, generate visualizations, “hacking”
- Processing.org (https://processing.org/)
  - Great examples, easy graphics, built-in editor, Java-based
- Python
  - “Massive” user base - tutorials, libraries, simplicity
  - Pre-installed on OSX and Ubuntu, Anaconda for Windows

An aside: Version Control Software

- Save early and save often!
- Share your code, use shared code, collaborate
- Git (https://github.com/)



Python, using Mu (“Moo”)

- Mu
  - Installs Python and some common libraries
  - Provides a text editor and terminal window
  - Can be used for standard Python 3, pygame Zero, and CircuitPython
- Let’s take a look!

Python 3 in Mu

- First, **run Mu in Python 3 mode**
- **Press the REPL button to run Python dynamically**
  - Read-Eval-Print-Loop interaction, like MATLAB

Python in REPL

- **Try it out!**
  - Make a variable
  - Do some math
    - Are there any issues with division?
  - Make a list
    - How do you reference an element?

Python

		Built-in Functions		
abs()	divmod()	input()	open()	staticmethod()
all()	enumerate()	int()	ord()	str()
any()	eval()	isinstance()	pow()	sum()
basestring()	execfile()	issubclass()	print()	super()
bin()	file()	iter()	property()	tuple()
bool()	filter()	len()	range()	type()
bytearray()	float()	list()	raw_input()	unichr()
callable()	format()	locals()	reduce()	unicode()
chr()	frozenset()	long()	reload()	vars()
classmethod()	getattr()	map()	repr()	xrange()
cmp()	globals()	max()	reversed()	zip()
compile()	hasattr()	memoryview()	round()	__import__()
complex()	hash()	min()	set()	
delattr()	help()	next()	setattr()	
dict()	hex()	object()	slice()	
dir()	id()	oct()	sorted()	

Python Conditional Statements and Indentation

Example, note the tabs and colons:

```
if a < 4:
    print(‘less than 4’)
else:
    print(‘not less than 4’)
```

- **Write an If statement**

Python

Operation	Meaning
`<`	strictly less than
`<=`	less than or equal
`>`	strictly greater than
`>=`	greater than or equal
`==`	equal
`!=`	not equal
`is`	object identity
`is not`	negated object identity

Python Libraries

- Mu comes with some common libraries (note for Nick: ...\AppData\Local\Mu\pkgs)
- Example: NumPy
  - Import numpy as np
  - np.mean(), np.std(), ...
  - https://docs.scipy.org/doc/numpy-1.13.0/reference/
- **Take the standard deviation of a list**

How To Write and Run a Program

- Up in the editor where it says # Write your code here :-)
  - Code that appears after # is a comment and ignored
  - Mu will autocomplete if it can guess a name

Make a .py and Test

- Make a .py file
- **Create some variables, do some math, print() the results**
- **Press the Run button to run the code!**
- **Press the Stop button when done**

Write a Program

- Use the function input() to get a number from the user
  - Example: temperature = float(input('What is the temperature? '))

- **Write a program that asks the user for the current temperature, and if it is above 70, print “Shorts weather!” and otherwise print “Brrrrrr!”**

Write Your Own Function in Python

- For readability and modularity, collect your code into functions
- Example:

```
def doubleIt(varIn):
    return varIn*2
def main():
    a = 2
    b = doubleIt(a)
    print(b)
main()
```

- **Write a function that takes a list and returns the average**

Plotting Data

- **Use matplotlib to make a MATLAB-like plot**

```
import matplotlib.pyplot as plt
...
plt.plot(x,y,'ro-')
plt.show()
```

- https://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.plot

While Loop

- Continue doing an action while something is true
- Example

```
a = 5
while a > 0:
    print(a)
    a = a - 1
```

- **Write a loop that continues as long as the user types in a number greater than 0**

For Loop

- Perform an action a set number of times
- Use the function range() to set the number of times
- Example: (**note the start value**)

```
for x in range(5):
    print(x)
```

- Example:

```
c = [1,2,3,4]
for k in range(len(c)):
    print(c[k])
```

Importing Data from a File

- Create a new file called data.csv in your working folder
- Make two columns of data separated by commas
- **See if Python can read the file**:

```
import csv
with open('data.csv', 'rb') as f:
    reader = csv.reader(f)
    for row in reader:
        print row
```

Store the Data into Lists

- Before you open the file, make two blank lists

  x = []

  y = []
- Use the append() function to add the data from each row into each list

  x.append(row[0])
- **Print each list to check if you've saved the data**

Lots of Data

- Rename the CTA Ridership CSV file something simple, like cta.csv, and move it to your working folder (mu_code)
- Take a look at the row structure
- **Which station is busier, Foster or Noyes?**

Pygame Zero

- Draw objects, make sounds!

```
WIDTH = 500
HEIGHT = 100
TITLE = "Fading Green!"
c = 0

def draw():
    screen.fill((0, c, 0))

def update(dt):
    global c, HEIGHT
    c = (c + 1) % 256
    if c == 255:
        HEIGHT += 10

def on_mouse_down(button, pos):
    print("Mouse button", button, "clicked at", pos)
```

Pygame Zero

```
alien = Actor('alien')
alien.topright = 0, 10

WIDTH = 500
HEIGHT = alien.height + 20

def draw():
    #screen.clear()
    alien.draw()

def update():
    alien.left += 2
    if alien.left > WIDTH:
        alien.right = 0

def on_mouse_down(pos):
    if alien.collidepoint(pos):
        set_alien_hurt()

def set_alien_hurt():
    alien.image = 'alien_hurt'
    sounds.eep.play()
    clock.schedule_unique(set_alien_normal, 1.0)

def set_alien_normal():
    alien.image = 'alien'
```