In this course we are using Python 3. You can download it from:

https://www.python.org/downloads/

1.1 Python Syntax

Basic Syntax

- simple data structure: , []
- print
- simple algebraic operator: +, -, *, /, %, **
- Logical operator: and, or, not
- If statement, Loop Statement, and Indentation.

```python
def Example1(a, b):
c = a + b + 4  # use "#" for comments
c = "test"    # Python is weak typing programming language
if c == "test":
d = 10        # a local variable,
    # use INDENTATION to indicate this statement is inside "if"
c = 1        # not inside "if"
while (c < 10):
c +=1        # Note: python does not support ++
print(c)     # Question: What values are printed?
for i in range(10):
    print(i)    # Question: What values are printed?
for i in range(5,20,3):
    print(i)    # Question: What values are printed?
return c
```

File IO

- open()
- read()
- readlines()
- write()
def Example2(Input, Output):
    Fout = open(Output, 'w')
    with open(Input) as f:
        lines = f.readlines()  # lines stores strings from Input line by line
        for l in lines:  # iterate through all lines
            words = l.split()  # split strings by space
            Fout.write(words[-1])  # write the last word to Fout
            Fout.write('
')

Simple Object-Oriented Programming:

class GradStudent:
    advisor = None
    school = None
    courseList = []
    def __init__(self, advisor, school):
        self.advisor = advisor
        self.school = school
        self.courseList = []
    def switchAdvisor(self, advisor):
        self.advisor = advisor
    def takeCourse(self, course):
        self.courseList.append(course)

1.2 Useful Libraries

- SciPy [https://www.scipy.org](https://www.scipy.org)
- NumPy [https://www.numpy.org](https://www.numpy.org)
- MatPlot [http://matplotlib.org](http://matplotlib.org)

1.2.1 Numpy

Tutorial: [https://docs.scipy.org/doc/numpy-dev/user/quickstart.html](https://docs.scipy.org/doc/numpy-dev/user/quickstart.html)

One of Numpy’s main objects is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In NumPy dimensions are called axes. The number of axes is rank.

**ndarray**: The numpy’s array class is called ndarray, it’s not the same as the standard array.array, which only handles one-dimensional arrays and offers less functionality.

- ndim(): The number of axes of array.
- shape(): The dimensions of the array.
- size(): total number of elements
• array(): creating array

```python
import numpy as np
def Example3():
a = np.arange(15).reshape(3,5)
print(a)
print(a.shape)
print(a.ndim)
a = np.array([2,3,4])
```

1.2.2 SciPy

**Tutorial:** [https://docs.scipy.org/doc/scipy/reference/tutorial/](https://docs.scipy.org/doc/scipy/reference/tutorial/)
SciPy is a collection of mathematical algorithms and convenience functions built on the Numpy extension of Python. You can use SciPy to solve the following problems:

1. Integration
2. Optimization
3. Interpolation
4. Fourier Transforms
5. Singal Processing
6. Linear Algebra
7. Statistics
8. Image Processing
9. File IO
10. Spatial data structures and algorithms

```python
import numpy as np
def Example4():
    for i in range(20):
        noise = np.random.laplace(scale=5)
        print(noise)
```

1.2.3 MatPlot

**Tutorial:** [http://matplotlib.org/users/tutorials.html](http://matplotlib.org/users/tutorials.html)
Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

**Bar chart Sample Code:** [https://matplotlib.org/examples/api/barchart_demo.html](https://matplotlib.org/examples/api/barchart_demo.html)