Week 2: Basics of Python

Computational Tools and Techniques in STEM

Feb 5-7, 2019

Learning Goals

- **L1**: Getting started with Spyder.
- **L2**: If statement, for loop, functions and scoping.
- **L3**: Quick introduction to Sympy.
- **L4**: Importing user-defined functions from another file.
- **L5**: Plotting using Matplotlib.
- **L6**: Reading and writing data.
- **L6**: Project exercise.

Plotting in Matplotlib

(a) Create two numpy arrays, xdata and ydata.
(b) Import numpy and matplotlib.pyplot modules.
(c) Let's get plotting!
(d) Add legend.
(e) Add figure title.
(f) Add axis labels.
(g) Change the font size and color.
(h) Change the plot color and marker.
(i) Change the line thickness.
(j) Add math symbols to the axes.

Matplotlib Gallery: `https://matplotlib.org/gallery.html`

Plotting Exercise

```python
import numpy as np
import matplotlib.pyplot as plt

# Data to be plotted
xdata = np.linspace(0, 1, 20)
ydata = np.linspace(0, 1, 20)

# Plotting the data
plt.plot(xdata, ydata, "Results")
plt.title("Sample plot", fontsize=20)
plt.xlabel("Price", fontsize=20)
plt.ylabel("Quantity", fontsize=20)
plt.xticks(fontsize=20)
plt.yticks(fontsize=20)
plt.legend(fontsize=20)
plt.savefig("test.png") # save plot to a file
plt.show() # displays the plot on screen
```

Reading and Writing Data

```
import numpy as np

# open/create a file and append to it
f = open('sample.dat', 'a+b')

# create a numpy array
my_arr = np.zeros(3)

# fill the array with some values
for i in range(0,len(my_arr)):
    my_arr[i] = i*2

# write data to file and close it
np.savetxt(f, my_arr)
f.close()

# read from file
read_data = np.loadtxt('sample.dat', usecols=0)
```

Exercise

Write a program that has two particles (spherical and cubical in shape) starting at origin and moving randomly. Some of the particle parameters are as follows:

- density (density = 1.05 kg/m^3 and 2.10 kg/m^3, respectively)
- length (radius = 200×10^{-6}m; cube length = 200×10^{-6}m)

You are required to calculate the following:

- Volume of the particles based on shape.
- Mass of the particles. (mass = volume * density)
- Force of gravity on the particle. (force = mass * gravitational acceleration)
- Move the particle randomly for 200 steps (xnew = xold + randomstep), updating the position at each step.
- Plot the positions at every 20th step.

Note: Additional particle properties and behavior will be gradually incorporated into this model.