Computational Astrophysics

E. Larrañaga

Observatorio Astronómico Nacional
Universidad Nacional de Colombia

December 24, 2018

Outline

Hello World!

```
# Print the Hello World! statment
print("Hello World!")
```

Importing a Module

```
# import the math module
import math

x = math.pi
print("An approximte value of pi is %8.5f" % x)
```

Print Formatting

```
%[width][.precision]type
```

Print Formatting

```
%[width][.precision]type

[width] : total number of digits (including the decimal point)
[.precision] : digits in the decimal part of the number
```

Print Formatting

```
%[width][.precision]type

[width] : total number of digits (including the decimal point)
[.precision] : digits in the decimal part of the number

type:
d (Integer)
f (Float)
e (Scientific Notation)
g (Same as "e" if exponent is greater than -4 or less than precision,
"f" otherwise)

...
```

Importing Modules I

```
# import the sys and math modules
import sys , math

x = math.pi / 2.0
print(math.sin(x))

sys.exit()          # leave program here

print("This will NOT print!!")
```

Importing Modules II

```
# import the sys and math modules
import sys
import math as m

x = m.pi / 2.0
print(m.sin(x))

sys.exit()        # leave program here

print("This will not print!!")
```

Importing Modules III

```
# import the sys and math modules
import sys
from math import pi, sin

x = pi / 2.0
print(sin(x))

sys.exit()        # leave program here

print("This will not print!!")
```

Importing Modules IV

```
# import the sys and math modules
import sys
from math import *

x = pi / 2.0
print(sin(x))

sys.exit()          # leave program here

print("This will not print!!")
```

Simple Math I

```
import math as m

# Simple math operations
a = 2.0
b = 1.0
c = a + b
d = c**2  # square c
e = m.sqrt(c)  # take the sqrt of c
f = d/c  # divide d by c

# Print results
print(a,b,c)
print(d)
print(e)
print(f)
```

Simple Math II

```
import math as m
# Trigonometry
pi = m.pi
theta = 2*pi
print(m.sin(theta))
print(m.cos(theta))
print(m.tan(theta))

# Other Operations
a = 5.0
print(m.exp(a))
print(m.log(a))
print(m.sinh(a))
```

Numpy Arrays I

```
import numpy as np

a = np.array([0.0, 1.0, 2.0, 3.0])
b = np.arange(10)
c = np.zeros(10,float)
d = np.zeros(10,int)
e = np.ones(10)
f = np.random.random(10)

print(a)
print(b)
print(c)
print(d)
print(e)
print(f)
```

Numpy Arrays II

```
import numpy as np

a = np.array([[1, 2, 3], [4, 5, 6],[7, 8, 9]])
b = np.array([np.arange(3), np.arange(3),np.arange(3)])
c = np.zeros([3, 3],float)
d = np.ones([3, 3])
e = np.random.random([3, 3])

print(a)
print(b)
print(c)
print(d)
print(e)
```

Numpy Arrays Functions

```
import numpy as np

a = np.random.random([3, 4])

print(a)
print(np.ndim(a))    # Number of dimensions
print(len(a))          # Size of the first dimension
print(len(a[0, :]))    # Size of the second dimension
print(a.max())         # Max entry
print(a.min())          # Min entry
print(a.sum())         # Sum of all the elements
print(a[0,:].sum())  # Sum of first row
print(a[:,0].sum())  # Sum of first column
```

Conditional Statements and Indentation I

```
import numpy as np

# get a random number in the range [0.0,1.0)
a = np.random.random()

if a < 0.5:
    print ("a = %5.6g is < 0.5!" % a)
    print ("Small a!")
else:
    print ("a = %5.6g is >= 0.5!" % a)
    print ("Large a!")
```

Conditional Statements and Indentation II

```
import numpy as np

# get a random number in the range [0.0,1.0)
a = np.random.random()

if a < 0.3:
    print("a = %5.6g is < 0.3 " % a)
    print("Small a!")
elif a>= 0.3 and a<0.6:
    print("0.3 <= a = %5.6g < 0.6 " % a)
    print("Medium a!")
else:
    print("a = %5.6g is >= 0.6 " % a)
    print("Large a!")
```

Loops I

```
for i in range(100):
    print(i)
```

Loops II

```
for i in range(1,100,2):
    print(i)
```

Loops III

```
i = 0
while i<100:
    print(i)
    i = i + 1
```

Functions I

```
def myfunction(x):
    return x**2 + 5

a = 2.0
print(myfunction(a))

print(myfunction(3.5))
```

Functions II

```
def myfunction(x,y):
return x**2, x + y

a = 2
b = 3

c,d = myfunction(a,b)

print(c)
print(d)
```

Reading Files I

```
# Open the file
infile = open("example_file1.txt","r")

# Read the data
indata = infile.readlines()

# Close the file again
infile.close()

print(indata)
```

Reading Files II

```
import numpy as np

# Open the file
infile = open("example_file2.txt","r")

# Read the data
indata = infile.readlines()

# Close the file again
infile.close()

print(indata)
```

Reading Files II (cont.)

```
# get number of data lines in the file
# (-1, because first line is a comment)
n = len(indata) - 1

# allocate two numpy arrays
time = np.zeros(n)
data = np.zeros(n)
```

Reading Files II (cont.)

```
# parse the data (separated by a comma in the file)
for i in range(n):
    splitline = indata[i+1][:-1].split(',')
    time[i] = float(splitline[0])
    data[i] = float(splitline[1])

print(time)
print(data)
```

Writting Files

```
import numpy as np

# open the file for writting
outfile = open("outfile.txt","w")

# how many lines do we want to output
n = 100

# write a header
headerstring = "# This is header for the data\n"
outfile.write(headerstring)

# write some random data with a formatting
for i in range(n):
    ran1 = np.random.random()
    ran2 = np.random.random()
    outstring = "%10.10E,%10.10E\n" % (ran1,ran2)
    outfile.write(outstring)

#close the file
outfile.close()
```

Reading and Writting Files with Numpy

```
import numpy as np

# read file
data = np.loadtxt("example_file3.txt",comments="#")
print(data)

# do something with data, e.g., square each entry
data[: , :] = data [: , :]**2
print(data)

# write result
np.savetxt("outfile3.txt",data,fmt="%10.10E",\
    header="This is a data file")
```