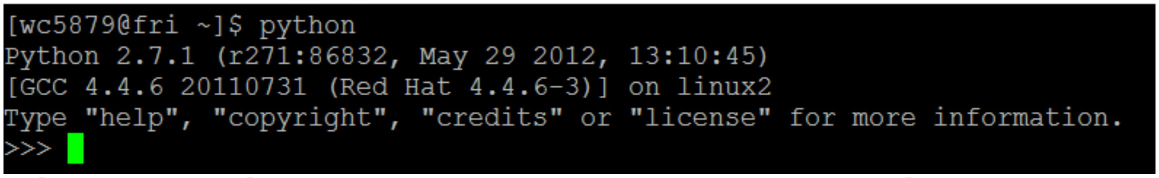
Computational Materials Python Cheat Sheet

This cheat sheet was created in python’s interactive mode. In the interactive mode you type commands into the interpreter and directly find out the results. The “>>>” is the prompt used to let you know that python is ready to accept a command. To start python in interactive mode type **python** on command line. It will look as follows:



```
[wc5879@fri ~]$ python
Python 2.7.1 (r271:86832, May 29 2012, 13:10:45)
[GCC 4.4.6 20110731 (Red Hat 4.4.6-3)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Below are examples of how to use the python programming language.

Variable Assignments

>>> x=5 >>>x 5	Declaring an integer variable x with value 5 or assign variable x to the integer 5

Types of Variables

>>> name = 'Bob'	String
>>> I=5	Integer
>>> F=5.0	Float
>>> happy='True'	Bool

Calculations with variables or arithmetic operators

>>>y=7	
>>>y+5 12	Sum of two variables
>>>y-5 2	Subtraction of two variables
>>>y*2 14	Multiplication of two variables
>>>y**2 49	Exponential of two variables
>>>y%2 1	Remainder of a variable
>>>y/14 0	Division of two ***integers***
>>>y/14.0 0.5	Division with *at least* one float

Type Conversion or Type Casting

>>>str('5.1') 5.1	Variables to strings
>>>int(5.0) 5	Variables to integers
>>>float(5) 5.0	Variables to float
>>> type(5.4) <type 'float'>	Returns variable type

Lists

>>>Names = ['bob', 'finn', 'ray']	Note python starts its indexing/numbering at 0 not 1
>>>Names	
['bob', 'finn', 'ray']	
>>>Names[0]	Select item at index 0
‘bob’	
>>>Names[−1]	Select last item in the list
'ray'	
>>>Names[2]	Select third item in list
'ray'	
>>>len(list)	Returns length of list
3	

Boolean Operators

>>> x=5	
>>> x==5	Equality: are the two values equal?
True	
>>> x!=5	Inequality: are the two values not equal?
False	
>>> x>5	Greater than
False	
>>> x<=5	Less than or equal to
True	
>>> True and True	Are both values True?
True	
>>> True and False	
False	
>>> True or True	Are either values True?
True	
>>> True or False	
True	

If Statements (Recall indentation rules in python!)

>>> x=5	
>>> if x==5:	If conditional statement is True
... print True	then do
...	
True	
>>> if x==5:	If conditional statement is True
... print True	then do
... else:	otherwise
... print False	do
...	
True	
>>> if x==2:	If the conditional statement is True
... print 'value = 2'	do this
... elif x<2:	else if this conditional statement is True
... print 'value < 2'	do this
... else:	otherwise
... print 'value > 2'	do this
...	
value > 2	

Loops

```
>>> x=0
>>> while x < 5:
...     print x
...     x += 1
...
0
1
2
3
4


>>> for i in range(0,5):
...     print i
...
0
1
2
3
4
>>>
```

```
While the variable x is less than 5,
     Print the value of x
     And add one to the value of x
```

Output from while loop

```
For i ranging from 0 until 5
     Print the value of i
```

Output from for loop

Defining functions

```
>>> def print_number(x):
...     print x
...
>>> print_number(4)
4

>>> def return_number_add_one(x):
...     return x+1
...
>>>
new_number=return_number_add_one(5)
>>> print new_number
6
>>>
```

Define the function which prints the input value, x.

Call the function with input value x=4.

Define a function which *returns* the input value, x, plus one

Call the function and set the output equal to new_number

Importing modules – Numpy

```
>>> import numpy
>>> numpyarray =
numpy.array([1,2,3],float)
>>> numpyarray
array([ 1.,  2.,  3.])
>>> numpy.mean(numpyarray)
2.0
```

Import module (numpy)
Create a numpy array of floats called numpyarray

Use numpy to calculate the mean of the values in numpyarray

If you want to find more numpy functions, try googling numpy + (insert function your need!).