Lab Questions: Lab Session 2

Deadline: 30.08.2017 11:59pm SGT

Complete all assignments below. Create a file <yourMatricNo_Lab2>.py and use it to save your answers (write the commands, not what the console displays after executing your commands) to those questions below that are marked with an asterisk *. These questions are 8, 10 and 11. Once you are done with it, submit the file via `iNTU`.

Important!!! Please name the file according to the requirements, and upload each file separately and not in a Zip file or similar. The submission system closes at the deadline. Hence after that, you will get no marks for your solution.

1. Start an interactive Python session and type a simple printing command (for example display the string *Singapore*) and a simple addition command (for example add 12 to 15) in the Python console. Then, write the very same commands in a Python script and run this script. Is there a difference ?

   Solution:
   Yes: the addition result is not displayed when launched from a script (while it is in the interactive mode). Remember that Python has two basic modes: normal and interactive. The normal mode is where your scripts (.py files) are fed to the Python interpreter. In that mode, you have to use the function `print` to display something in the console (in our case, we should have written `print(12+15)`). In contrary, the interactive mode gives immediate feedback for each statement and an expression result is automatically printed in the console.

2. Create a variable to store the atomic weight of Lithium (6.941)

   Solution:
   ```
   >>> lithium = 6.941
   ```

3. Create a variable `myage` and store your age in it. Subtract 5 from the value of the variable. Add 8 to the value of the variable. Print the final value contained in `myage`.

   Solution:
   ```
   >>> myage = 33
   >>> myage = myage - 5
   >>> myage = myage + 8
   >>> print(myage)
   36
   ```

4. The function `help()` is very useful when you want to find information about a function, a Python module, etc. To try it out, import the `math` module and use the function `help` to display informations about this module (`help(math)`). Then, using this help:

   (a) find and use the function from the `math` module that computes the hyperbolic sine of an angle in radian.

Solution:
```
>>> help(math)
>>> help(math.sinh)
>>> math.sinh(1)
1.1752011936438014
```

(b) find and use the function from the `math` module that converts degrees into radians. Computes the sine of an angle in degree using that function and the `math.sin` function. Verify for example that with 90 degrees you obtain 1.

Solution:
```
>>> help(math)
>>> help(math.radians)
>>> math.sin(math.radians(90))
1.0
```

5. Each type in Python has specific attributes and methods that can be accessed using the dot (.) operator (to simplify, view attributes and methods as functions). For example, create a variable `my_float` and initialise it with a float value. On Spyder, you can see the list of attributes and methods available for that variable by typing `my_float.` and then pressing the Tab key on your keyboard. Alternatively, you can list it by typing the command `dir(my_float)`. For example, the float type has a method `as_integer_ratio()` that decomposes the float as a ratio of two integers: `>>> my_float=0.25`
```
>>> my_float.as_integer_ratio()
(1, 4)
```

(a) Explore the attributes and methods of the string type and then find and use the method that converts your string to uppercase.

Solution:
```
>>> help(str.upper)
>>> my_string = "Hello world"
>>> my_string.upper()
'HELLO WORLD'
```

(b) What is the method `split` doing ?

Solution:
```
>>> help(str.split)
Return a list of the words in S, using sep as the delimiter string.  If maxsplit is
given, at most maxsplit splits are done.  If sep is not specified or is None, any
whitespace string is a separator and empty strings are removed from the result.

>>> my_string = "Hello my name is John"
>>> my_string.split()
[ 'Hello', 'my', 'name', 'is', 'John' ]
```

6. The combined resistance R_T of three resistors R_1, R_2 and R_3 in parallel is given by

$$R_T = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}}.$$

Create variables for the three resistors and store values in each, and then calculate the combined resistance.

Solution:
```
>>> r1 = 3
>>> r2 = 2.2
>>> r3 = 1.5
>>> rt = 1/(1/r1 + 1/r2 + 1/r3)
```

7. A vector can be represented by its rectangular coordinates x and y or by its polar coordinates r and θ. The relationship between them is given by the equations:
x = r * cos(θ)
y = r * sin(θ)
Assign values for the polar coordinates to variables `r` and `theta`. Then, using these values, assign the corresponding rectangular coordinates to variables x and y.

Solution:
```
>>> r = 5
>>> theta = 0.5
>>> x = r * math.cos(theta)
>>> y = r * math.sin(theta)
```

8. * A farmer grows tomatoes and carrots and sells them at a market. Currently the tomato price per kilo is 3.29 SGD, and the carrots can be sold at 1.8 SGD per kilo. Assign the prices to variables named tomato_price and carrot_price. The farmer produced 123kg of tomatoes and 247kg of carrots. Calculate the following by typing <u>one</u> command (aka a single line):

   (a) Compute the amount of money received by the farmer if he sells all its tomatoes and carrots

   (b) Compute the amount of money received by the farmer if he sells 87% of its tomatoes and 67% of its carrots

   (c) The farmer has to pay 15% tax on his earnings from (b). Compute the amount of money received by the farmer after tax

   (d) Compute the same as (c), and round the total cost to the nearest dollar

9. Wind often makes the air feel even colder than it is. The *Wind Chill Factor* (WCF) measures how cold it feels with a given air temperature T (in degrees Fahrenheit) and wind speed (V, in miles per hour). One formula for the WCF is:

$$WCF = 35.7 + 0.6T - 35.7(V^{0.16}) + 0.43T(V^{0.16})$$

Create variables for the temperature t and wind speed v, and then using this formula calculate the WCF.

Solution:
```
>>> t = 20
>>> v = 11
>>> wcf = 35.7 + 0.6*t - 35.7*v**0.16 + 0.43*t*v**0.16
```

10. * Write a Python program to compute the following expressions (assume that the angles are in radian):

$$\textbf{a)}\quad \frac{(e^{1.4} + \ln(465^2))}{\sqrt{2}+14} + \frac{12}{\sqrt{e+4}}$$

$$\textbf{b)}\quad (-2.6)^{0.2} + \frac{e^{-\sqrt{43.3}}}{\tan(276)} + 17^{-1/7}$$

$$\textbf{c)}\quad \frac{\pi^3 - 5.6^2 + 1}{1.2^{\pi/2} - \sin(43)} + \left(\frac{14.8}{5}\right)^{\pi-1.8}$$

11. * Using the Python module `random`, generate a random
    (a) real number in the range from 0 (included) to 1 (not included)
    (b) real number in the range from 0 (included) to 0.2 (not included)
    (c) real number in the range from 0.5 (included) to 50 (not included)
    (d) integer in the range from 32 to 60 (included)

12. Using the Python module `random`, generate random numbers. Then, seed the random number generator with a certain value and observe that you still obtain new random values. Then, seed again the random number generator with the same value and observe that now the random values are repeating. Eventually, seed the random number generator with a new value to obtain new random values.

Solution:
```
>>> random.random()
0.9424502837770503

>>> random.random()
0.7398985747399307

>>> random.seed(4)
>>> random.random()
0.23604808973743452

>>> random.random()
0.1031660342307158

>>> random.seed(4)
>>> random.random()
0.23604808973743452

>>> random.random()
0.1031660342307158
```

```
>>> random.seed(5)
>>> random.random()
0.23796462709189137

>>> random.random()
0.5442292252959519
```