Laboratory 3: working with lists

Prerequisites: as in lab2, +lists, +iterating, if-else

Contents

PEP8

To monitor the coding style we turn on the PEP8 style guide monitoring in Spyder by:

1. Going to preferences in Spyder menu
2. Clicking on `Editor` in the selection list on the left
3. Clicking on `Code Introspection/Analysis` (top right corner)
4. Ticking the box `Real-time code style analysis`
5. Clicking `Apply` and `OK` (bottom right corner)

Exercises

Implement the following functions in a file called `lab3.py`:

1. A function `degree(x)` that takes an argument `x` in radian and returns the corresponding value in degrees. I.e. given a value `x`, the function should return

   $$x\,\frac{360}{2\pi}$$

   Example:

   ```
   In [ ]: degree(math.pi)
   Out[ ]: 180.0
   ```

2. A function `min_max(xs)` that computes the minimum value `xmin` of the elements in the list xs, and the maximum value `xmax` of the elements in the list, and returns a tuple `(xmin,xmax)`.

   Example:

   ```
   In [ ]: min_max([0, 1, 2, 10, -5, 3])
   Out[ ]: (-5, 10)
   ```

3. A function `geometric_mean(xs)` that computes the geometric mean of the numbers given in the list `xs`. Hint: Remember that `a**b` computes a^b (i.e. takes a-to-the-bth-power).

   Example:

   ```
   In [ ]: geometric_mean([1, 2])
   Out[ ]: 1.4142135623730951
   ```

4. A function `swing_time(L)` that computes and returns the time T [in seconds] needed for an idealized pendulum of length `L` [in meters] to complete a single oscillation, using the equation

   $$T = 2\pi\sqrt{\frac{L}{g}} \quad \text{with} \quad g = 9.81\,\mathrm{m/s^2}$$

   Example:

   ```
   In [ ]: swing_time(1)
   Out[ ]: 2.0060666807106475
   ```

5. A function `range_squared(n)` that takes an non-negative integer value `n` and that returns the list `[0, 1, 4, 9, 16, 25, ..., (n-1)^2]`. If `n` is zero, the function should return the empty list.

   Example:

   ```
   In [ ]: range_squared(3)
   Out[ ]: [0, 1, 4]
   ```

6. A function `count(element,seq)` that counts how often the given element `element` occurs in the given sequence `seq`, and returns this integer value. For example, `count(2,list(range(5)))` should return `1`.

   Example:

   ```
   In [ ]: count('dog',['dog', 'cat', 'mouse', 'dog'])
   Out[ ]: 2

   In [ ]: count(2, list(range(5)))
   Out[ ]: 1
   ```

Remember to check all functions for correctness, in particular: are the (i) functions tested, (ii) documented, (iii) have the right name, (iv) and does the file execute silently and without errors. Then submit `lab3.py` by emailing it to feeg1001@soton.ac.uk with the subject lab 3 for automatic assessment of this laboratory session. If some of the tests fail, you can improve your code and re-submit `lab3.py` as many times as you like.

Return to Top

Page last modified Sun Oct 28 15:20:57 2018. |