

Lab 4: iteration with for loops, random lists, argmin (09.23.19) CS103 Fall 2019

author john k johnstone jkj at uab dot edu

course CS103 Fall 2019

license MIT

version Fall 2019

materials

- lab04_19fa103.pdf (this document)
- lab04_19fa103.py (with functions you will implement)
- reverse_stubbed.py
- min_stubbed.py

lab partner

Please work with the lab partner that you chose or were assigned in Lab02.

purpose

Today you will learn how to generate large amounts of test data using the random module, then compute a better version of the minimum function. Notice that the random lists you generate may be used to test your min functions.

- practice iteration using for loops
- practice writing functions
- learn about the random module
- learn how to compute minimum and maximum properly using a for loop

preparing for Lab04

- set up your screen with Canopy terminal on left and Canopy editor on right
- remove the bottom panel of the Canopy editor
(the one that might tempt you to run your code there)
- download the Canvas materials and put in your 19fa103/lab/lab04 directory
- open the Python documentation for the random module, which is at
<https://docs.python.org/3/library/random.html>
- have the pdf of lecture07 and lecture08 available for reference
(for when you are in need of a refresher on iteration with for loops)

in-class example

Try this first, then sometime later in lab, one of the TA's will solve for the class at the board.

- compute the *maximum* of a list of ints, using a for loop (myMax)
(complete with docstring and test calls)

exercises (with your lab partner)

For each of these exercises, write a function, complete with docstring and test calls.

There may be more exercises here than you can complete today in lab: that is purposeful to keep you pleasantly busy, and give you more exercises outside lab.

- copy `reverse_stubbed.py` to `reverse.py` and practice implementing this function
- copy `min_stubbed.py` to `min.py` and practice implementing this function
- explore the Python documentation for the `random` module, paying particular attention to the functions `random`, `randint`, `randrange`; make at least 2 calls of each in your Python interpreter, or in a new file
- build a *random list* of n ints in the interval $[a, b]$ using a for loop (`randomListInt`)
- compute the *index of the minimum* of a list of ints, using a for loop (`argmin`)
- (ding, ding! the winner!)
explain why the index of the minimum element is more useful than the minimum element (insert as a comment in `lab04_19fa103.py` at the designated place)
- (**why argmin**)
give an example of a computation for which the index of the min is useful but the min is useless (insert as a comment in `lab04_19fa103.py` at the designated place)
- compute the *index of the maximum* of a list of ints, using a for loop (`argmax`);
how many lines of the function body change from `argmin`? how many characters?
- test your *myMax*/*myMin*/*argmin*/*argmax* functions using the random lists from *randomList*
- solve `reverse_stubbed.py` again: how fast can you get?
- solve `min_stubbed.py` again: how fast can you get?
- build a *random list* of n floats in the interval $[0, 1]$ using a for loop (`randomListFloat1`)
- build a *random list* of n floats in the interval $[a, b]$ using a for loop (`randomListFloat2`)

challenges (optional for A+)

- (monkeys tapping at a keyboard)
fn to build a *random list of strings* over the English alphabet a-z, using a for loop
- fn to build a *random triangle* in 3-space inside the unit cube, as a tuple of three 3-tuples (points in 3-space); define the unit cube here as $[-1, 1] \times [-1, 1] \times [-1, 1]$

deliverables

B attendance (that is, full participation throughout lab) and successful completion of the solved in-class problem

A attendance and successful completion of the solved in-class problem, *argmin*, *why argmin*, *randomListFloat2*

A+ attendance and successful completion of
the solved in-class problem, argmin, why argmin, randomListFloat2, and *one of the challenges*