

CITS5501 Software Testing and Quality Assurance
CITS5501
Semester 1, 2018
Workshop 1

Testing in Python

Refer to <http://docs.python-guide.org/en/latest/writing/tests/>, and the Python library documentation, for details on the libraries used here.

1. Write a python class called Box. The class is designed to represent a rectangle with a bottom-left and top-right corner specified as integer Cartesian coordinates.



The class should have:

- a constructor
 - a method called “area”, which returns the area, and
 - a method called “intersection” which, when given another Box, returns a new Box that is the intersection of the two. If the intersection is empty, it should return a box with both corner coordinates set to (0,0).
2. Write a basic *doctest* for each method in the class, and ensure the doctest runs correctly.
 3. Use unittest to build a simple set of test cases for this class.
 - How should the tests be organised?
 - How many tests do we need?

4. Are there any details in the specification in question (1) which are missing, or ambiguous? What are they? How you have you handled this in your code.

Compare your answers with other students. Have they found things you did not? Are there any questions you could have asked which might have highlighted those missing details or ambiguities?

5. What preconditions and postconditions would you give for the above methods? Are there any invariants which you think should hold of a `Box` object?

Examine the documentation for the Python *Hypothesis* testing library (at <https://hypothesis.readthedocs.io/en/latest/index.html>), and try to develop several tests using the library.

6. How do your tests from `doctest`, `unittest` and `Hypothesis` compare? Are there any advantages you can see of one over the other?