http://www.cs.cornell.edu/courses/cs1110/2018sp

Lecture 6: Specifications & Testing  
(Sections 4.9, 9.5)  
CS 1110 
Introduction to Computing Using Python 

[E. Andersen, A. Bracy, D. Gries, L. Lee, S. Marschner, C. Van Loan, W. White]
Representative Tests Example

```python
def last_name_first(full_name):
    """Returns: copy of full_name in form <last-name>, <first-name>
    full_name has the form <first-name> <last-name>
    with one or more blanks between the two names"
    end_first = full_name.find(' ')
    first = full_name[:end_first]
    last = full_name[end_first+1:]
    return last+', '+first
```

Representative Tests:
* last_name_first('Maya Angelou') Expects: 'Angelou, Maya'
* last_name_first('Maya Angela') Expects: 'Angela, Maya'

Unit Test: A Special Kind of Script

- A unit test is a script that tests another module. It:
  * Imports the module to be tested (so it can access it)
  * Imports `cornellasserts` module (for testing)
  * Defines one or more test cases that each include:
    * A representative input
    * The expected output
  * Test cases use the `cornellasserts` function:

```
def assert_equals(expected, received):
    """Quit program if expected and received differ"
```

Testing last_name_first(full_name)

```python
import name # The module we want to test
import cornellasserts # Includes the tests

# First test case
result = name.last_name_first('Maya Angelou')
cornellasserts.assert_equals('Angelou, Maya', result)

# Second test case
result = name.last_name_first('Maya Angela')
cornellasserts.assert_equals('Angelou, Maya', result)

print('All tests of the function last_name_first passed')
```

Organizing your Test Cases

- We often have a lot of test cases
  * Need a way to cleanly organize them

**Idea**: Bundle all test cases into a single test!

- High level test:
  * One of these for each function you want to test
  * High level test performs all test cases for function
  * Also uses some print statements (for feedback)

Bundling all the tests into a single test

```python
def test_last_name_first():
    """Calls all the tests for last_name_first""
    print('Testing function last_name_first')
    # Test 1
    result = name.last_name_first('Maya Angelou')
cornellasserts.assert_equals('Angelou, Maya', result)
    # Test 2
    result = name.last_name_first('Maya Angela')
cornellasserts.assert_equals('Angelou, Maya', result)

    # Execution of the testing code
test_last_name_first()
    print('All tests of the function last_name_first passed')
```

Debugging with Test Cases (Question)

```python
def last_name_first(full_name):
    """Returns: copy of full_name in the form <last-name>, <first-name>
    full_name has the form <first-name> <last-name>
    with one or more blanks between the two names"
    space_index = full_name.find(' ')
    first = full_name[:space_index]
    last = full_name[space_index+1:]
    return last+', '+first
```

Which line is "wrong"?
- A: Line 1
- B: Line 2
- C: Line 3
- D: Line 4
- E: I do not know

- last_name_first('Maya Angelou') gives 'Angelou, Maya'
- last_name_first('Maya Angela') gives 'Angela, Maya'