More Docstring Testing

Introduction to Computer Programming

Dr. Paul Vrbik

November 10, 2018

Question

Write tests for the following function then implement it.

```
def ourmax(x:int, y:int) -> int:
    '''Return the larger of x and y.
    '''
```

Question

Write tests for the following function then implement it.

```
def indices(cs:str, subcs:str) -> List[int]:
    '''Return the indices in cs at which non-overlapping
    copies of subcs start. subcs is non-empty.

    >>> indices('A Coool pool look', 'oo')
    [3, 9, 14]
    '''
```

Question

Write tests for the following function then implement it.

```
def insert_after(xs:List[int], a:int, b:int) -> List[int]:
    '''Insert a after each occurrence of b in list xs.
    '''
```

Question

```
def average_grade(grades:List[List[object]]) -> float:
    '''Return the average grade for all the students in grades
    where the inner lists contain a student ID and a grade.
    >>> grades = [['998765', 70], ['111234', 90], ['444567', 83]]
    >>> average_grade( grades )
    81.0
    '''
```

Question

```
def student_ids(grades:List[List[object]]) -> List[str]:
    '''Return the student IDs for all the students in grades
    where the inner lists contain a student ID and a grade.
    '''
```

Question

```
def choose_chars(xs:str, ys:str, mask:str) -> str:
    '''Return a string where index i is xs[i] if mask[i]
    is 0 and ys[i] if mask[i] is 1.

    Note: xs, ys, and mask are all of the same length.
    Note: mask consists only of characters 0 and 1.
    '''
```

Unit Testing

```
>>> import unittest
>>> msg = 'abs():expected , received '
>>> class TestNumbers(unittest.TestCase):
...     def test_positive(self):
...         input = 8
...         expected = 8
...         result = abs(input)
...         self.assertEqual(expected, result,
...             msg.format(input, expected, result))
>>> unittest.main()
```

Next Time

1. More computer science.