

CSC108H Lecture 28

Dan Zingaro

November 16, 2012

Testing Dictionary Inversion

- ▶ Remember back to our dictionary-inversion function
- ▶ It returns a dictionary where values are lists
- ▶ The order of elements in these lists does not matter, but different list orderings make `==` return `False`
- ▶ In our tests for this function, we can't just compare the returned dictionary with the expected dictionary
- ▶ Instead, check that the keys are the same, and that values are the same irrespective of order
- ▶ Also, check that the function does not modify the dict parameter

Testing Functions that Return None

```
def merge_dict(d1, d2):  
    ''' (dict of ({object: int}, dict of {object: int})) -> NoneType  
    Add key/value pairs from d2 into d1. If a key from d2 already  
    appears in d1, the new value in d1 is the sum of the values. If a  
    key appears only in d1 or d2, then the new value in d1 is the  
    original value from the dictionary that contained this key.  
    d2 is unchanged.
```

- ▶ Define dicts d1 and d2
- ▶ Call `merge_dict(d1, d2)`
- ▶ Assert three things: `merge_dict` returned `None`, d1 is as expected, d2 is unchanged