

CS 150: Intro to OOP, Fall 2008

Programming Assignment 4

Due *via email* by 11:59pm on Wednesday, March 18, 2009

For this assignment, you must work individually. As usual, you are welcome to discuss *general* python syntax, but your design and coding should all be done individually. Please refer to the class syllabus or consult the instructor if you have any questions.

1. Exercises 6.15 and 6.16

Please submit both class definitions (one called `Set` for 6.15 and one called `SortedSet` for 6.16) in a file called `myset.py`.

Some things to keep in mind:

- I expect your sets to be able to support any string or numeric value being added.
- Make sure to use comments and meaningful variable names throughout the classes.
- It might be helpful to write a `__str__` method for your `Set` class, also, so that you can test that your code is working.
- Remember to test your code! Import your classes in python, create several sets and sorted sets, and add/remove numbers and strings to them to see if everything is added correctly and with no duplicates.

2. Extra Credit: Write the following methods for either your `Set` or your `SortedSet` class:

- (a) Implement an `__add__` function (overloading `+`) which takes two sets and returns the union of them in a new set.
- (b) Implement an `__eq__` function (overloading `==`) which take two sets and returns `True` if the sets are the same.
- (c) Implement an `xor(self, other)` function, which will be called by `set1.xor(set2)`. The function should return all elements which are either in `set1` or in `set2` (but NOT in both).