

CSI 32 LECTURE NOTES (Ojakian)

Topic 6: Classes in Python and C++

OUTLINE

PRIMER: 1.5, 2.8

TRANSITION GUIDE: 7.1, 7.2

1. Python Classes versus C++ Classes

1. Python Classes versus Completely Public C++ Classes

- (a) This is the Python way.
- (b) Can do this in C++ (**non-standard! but that is what we will do for now ...**)
PROBLEM 1. *Do a basic class in both languages which just has a single int as internal data which can be modified. Then try the follow experiments with it:*
 - i. *Make everything public in C++. Remove the access indicator and see what happens.*
 - ii. *Outside variable with the same name as internal variable name.*
 - iii. *From an object, try accessing data directly.*
 - iv. *Define an external function with the same name as a class method.*

2. Some details on Python versus C++ Classes

- (a) Usual Differences still apply:
 - i. Bodies: Python Indentation versus C++ Braces.
 - ii. Static versus dynamic typing
- (b) Python: Everything public
- (c) Python: “self” parameter; C++: not
- (d) Constructor/initializer:
 - i. Python: “init”.
 - ii. C++: method with same name as class
- (e) Passing an object to a function:
PROBLEM 2. *In both Python and C++ try passing an object, and changing the internal data to see if it persists after the function call.*
- (f) Operator Overloading: Basically just different syntax
PROBLEM 3. *Overload equivalence checking so that objects are equal if they have the same internal data values.*
In Python: `__eq__` (with self and other argument)
In C++: `operator==` (with one argument)

3. Bigger Class

PROBLEM 4. *In Python (then in C++) write a class for a general role-playing character (like “D and D”), called `RPGchar`. It should have two kinds of internal data: hit points (a non-negative integer) and alignment (a string that can take on the value ‘good’, ‘bad’, or ‘neutral’). It should have at least the following methods: 1) returns hit points, 2) return alignment, 3) damage self (lose hit points), 4) improve self (gain hit points)*

4. Inheritance

PROBLEM 5. *Do simple inheritance of first simple class.*

PROBLEM 6. *In Python (then in C++) write a child class for `RPG_Char` called `Fighter`. It has the following additional internal data: A stash of weapons (just represented as single word strings). Write methods to 1) add a weapon and 2) see the inventory, and 3) fling a weapon.*

The method fling removes the most recently added weapon, as it will be thrown. If the fighter tries to fling a weapon when she has none she losses some number of hit points (we’ll decide).