## CSCI 204: Data Structures & Algorithms **ADT: Operator**

Overloading

## Quick review: operator overloading

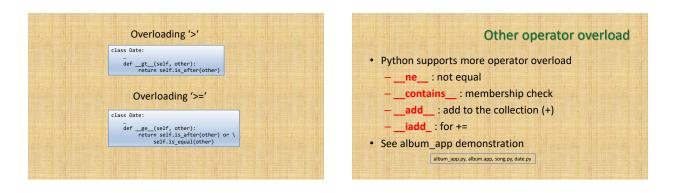
- We have learned some basic features of OOP, e.g. the Date class
  - Constructor: def \_\_init\_\_(self)
  - String representation: def \_\_str\_\_(self), or def \_\_repr\_(self)
  - Method within a class: def is\_leap\_year(self)
  - Object attributes (object variables ...) self.year, self.month, self.day.
- We will discuss and practice the topic of operator overloading

## What does it mean?

- An operator such as '==', '>' can be associated with a function to reflect its meaning.
- · E.g., in our Date class, we have three functions - is\_equal(), is\_before(), is\_after()

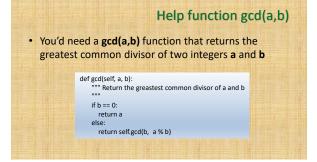
  - When comparing two Date objects, we'd say d1.is\_equal(d2), d1.is\_before(d2), d1.is\_after(d2)
- · If we implement operator overloads for the Date class, we could have said
  - d1 == d2, d1 < d2, d1 > d2

Overloa	ading '=='
class Date:	
	other.year and \ == other.month and \ other.day:
If the function is_equal() has been	n defined, we can do one of the following
class Date:	class Date:
_ defeq_(self, other):	defeq(self, other): return self.is_equal(other)



## **Build a Rational ADT**

- A rational is a fraction number such that both the enumerator and the denominator are integers, relatively prime to each other
- Build a Rational ADT such that
  - Support common arithmetic rational operations
  - x, y are two Rationals, x+y, x-y, x\*y, x//y are all Rationals
  - Support comparisons
  - x, y are two Rationals, x < y, x <= y, x == y, x >= y, x > y returns a True or False



	Examples of Rationa
	x = 2/3, y = 1/2
	x + y = 7/6
•	x - y == 1/6, y - x == - 1/6
•	x * y == 1/3
•	x // y == 4/3
•	x > y True
•	x >= y True
•	x == y False
•	x < y False
•	x <= y False
	Test program is on the course website, once finished your implementation, try it out.