

Object-Orientation

12-3-2010

Opening Discussion

- IcP solutions.
- Faster Quicksorts.

Object-Orientation

- We have been dealing with objects all semester, but we haven't really faced object-orientation head on.
- The OO paradigm is characterized by encapsulation, the grouping of data and functions together into objects.
- The data is called members and the functions are called methods.
- The idea is that an object knows some things and how to do some things.

Classes

- Scala is a class-based OO language. In the code we write classes which act as the blueprints of objects.
- These start just like the case classes we saw before, but the word `case` isn't required.
- Put the body of the class in curly braces after the declaration and arguments.

Differences from Case Classes

- Members are private by default so you can only see them in the class.
- Have to be made with `new`.
- Code in the body of the class is executed immediately.
- Functions defined in the body are methods of the objects.
- Data defined in the class are members of the objects.
- You can make things private.

Making Objects

- The class is only a blueprint. To get an object we have to instantiate an instance from the class.
 - `new ClassName(arguments)`
 - This expression can be assigned to values or passed into functions. The type is the name of the class.
- Once you have an object you can access members and methods using the dot notation.

Operators as Methods

- You can use symbols for method names and use them with operator syntax.
- This lets you do things like $a+b$ when a and b are of a type you created.

Minute Essay

- Questions?
- Interclass problem:
 - Write a class to represent either rational numbers or complex numbers. Give it appropriate methods for addition, subtraction, multiplication, and division.