Objects III

- You should have a dog class that supports energy. Playing fetch decreases a dog's energy, sleeping increases it.
- Start with your code from last time, or get feb-24-start.cpp.
- ***Warmup***: Fill in the `sleep` method. This method should let the dog sleep for a certain number of hours, and increases their energy by that amount.
- Fill in the `chase` method. Add a method `chase(dog & buddy)` to your class. This method will let your dog chase another dog. A dog can only chase another if both dogs' energies are above zero! Inside the method, print a message with both dogs' names. This method should decrease both dogs' energies by 1.
- ***Hint***: Inside a class, you have access to your own object's private variables, plus private variables of other objects that are passed in as arguments!

Constructors and destructors

- A ***constructor*** (abbrev **ctor**) is a method that is run automatically when an object is created.
- A ***destructor*** (abbrev **dtor**) is a method that is run automatically when an object is "destroyed."
  - For all objects right now, this means when the object goes out of scope.

Constructors

- Constructors are commonly used to initialize the fields (variables) in a class to appropriate values.
- Without constructors, the user would have to set all the fields in a class by hand after each object creation.
- The name of a constructor is always the same name as the class itself.

Dog default constructor

- What are appropriate values to initialize each field to in our dogs?

Dog constructors

- Classes can have multiple constructors.
- The default constructor never takes any arguments, but other constructors can.
- These arguments are typically used to set the fields of the class.

Destructors

- The name of a destructor is always the same name as the class, prefaced with a ~ (tilde).
  - Destructors never have any arguments, and there can be only one per class.