Inheritance I

CMSC 202

Warmup
Identify which constructor each of the following use (default, non-default, copy)

MyClass a;
MyClass b(a);
MyClass c(2);
MyClass* d = new MyClass;
MyClass* e = new MyClass(*d);
MyClass* f = new MyClass(4);

Code Reuse
How have we seen Code Reuse so far?
Functions
   Function Libraries
      Ex: math -> pow, sqrt
Classes
   Class Libraries
      Ex: vector, string
Aggregation
   Customer “has-a” DVD
   RentalSystem “has-a” Customer
Object Relationships

“Uses a”
Object_A “uses a” Object_B
Ex: Student sits in a chair

“Has a”
Object_A “has a” Object_B
Ex: Student has a name

“Is a” or “Is a kind of”
Object_A “is a” Object_B
Ex: Student is a kind of Person

Inheritance

What is Inheritance?
Unfortunately – not what your parents/grandparents will be giving you…

Inheritance
“is a” or “is a kind of” relationship
Code reuse by sharing related methods
Specific classes “inherit” methods from general classes

Examples
A student is a person
A professor is a faculty member
A lecturer is a faculty member

Inheritance Hierarchy
Why Inheritance?
Abstraction for sharing similarities while retaining differences
Group classes into related families
Share common operations and data
Multiple inheritance is possible
Inherit from multiple base classes
Not advisable
Promotes code reuse
Design general class once
Extend implementation through inheritance

Inheritance and Classes
Base class (or superclass)
More general class
Contains common data
Contains common operations
Derived class (or subclass)
More specific class
Inherits data from Base class
Inherits operations from Base class
Uses, modifies, extends, or replaces Base class behaviors

Inheritance Example
Inheritance
Assume the hierarchy on the right…
A is Base class
B is derived class
B derives from A
Every B is an A
Every A is NOT a B
Some A’s are B’s

Class A
   ▼
Class B
   ▼
A objects
   ▼
B objects

Inheritance
Assume the hierarchy on the right…
Everywhere an A can be used, a B can be used
Parameters
Return values
Items in vectors
Items in arrays
Reverse is not true…

Inheritance so far?
ifstream is an istream
ofstream is an ostream

Trip to the Zoo
Animal
  ◄
  ◄
  ◄
  ◄
  ◄
  ◄
  ◄
  ◄
Mammal
giveBirth()
reproduce()
Reptile
layEggs()
Lion
roar()
Dolphin
doTrick()
Rattlesnake
rattle()
Gecko
loseTail()
Inheritance

class BaseClass
{
    public:
        // operations
    private:
        // data
};
class DerivedClass : public BaseClass
{
    public:
        // operations
    private:
        // data
};

Indicates that this derived class inherits data and operations from this base class.

Inheritance in Action

class Animal
{ }
class Mammal : public Animal
{ }
class Lion : public Mammal
{ }
class Dolphin : public Mammal
{ }
class Reptile : public Animal
{ }
class Gecko : public Reptile
{ }
class Rattlesnake : public Reptile
{ }

Challenge

Draw the hierarchy for a Vehicle class

What kinds of vehicles are there?
  Personal, Commercial, etc.
What kinds of personal vehicles are there?
  Cars, Motorcycles, Trucks, etc.
What kinds of commercial vehicles are there?
  Planes, Trains, etc.