

COMP125 - Week 12 L1

Reuben Walker

November 5, 2014

Assignment Question Scaling for the map. How do we do this? Subtract the smallest value from all the numbers. This makes it 0. Then we need a multiplier so that they shrink to the size. The multiplier needs to be so that the difference between the highest and the lowest equals the furthestest point on the west.

Inheritance and Polymorphism Object Oriented Design is where we design our programs by thinking about the entities in our world. Entities are modelled as objects in Java. Methods provide operations on objects to get things done. This gives us a way of breaking down problems into meaningful parts. Each part is simpler, and easier to implement and test. We put the objects together to build the application. In COMP115, we were using procedural based design. drawFish, drawBarriers, etc. In OOD, we look at the problem and see how we can break it down into objects. Nearly every problem in computing is looked at by breaking things down into smaller bits. It's how computing is related to mathematics.

Example Problem We want to build a system to manage the results of the soccer matches in the local league. There are 10 teams, and they all play matches against the other teams through the season. For each match, we want to record the players in the team, the referee, the score, the number of fouls, and whether there were penalties during the match. We also need to keep track of the grounds that we play at. The entities are the nouns in here.

- League
- Team
- Match
- Ground

Inheritance I seem to have gotten distracted trying to sort out the scaling in my assignment.

Overloading vs Overriding Overloading is when we have one method name with different argument lists. Overriding is where we have the same method name and argument list in a derived class.

Encapsulation

Protected If something is protected, it's visible to derived classes, but invisible to everything else. It's halfway between private and public.

Package Access Package Access is where we leave off the private, public, or protected. Any class in the same package can access the method or instance variable.

Polymorphism