CompSci 6
Programming Design and Analysis

Nov 19, 2009
Prof. Rodger
Prof. Forbes
Announcements

• Read for next class, Chap. 16.2
• Reading Quiz for Tuesday, Dec 1
• Assignment 7 out
  – Do today’s classwork BEFORE doing this
• Class on Nov 24 is time to work on Assignment 7
Classwork Today – Recursive Art

• Two ways to draw art recursively
  – One object
    • Repeatedly draw the same object smaller
  – Multiple objects
    • Each object is “linked” to a smaller object
    • Each object draws itself
    • See the myNext variable
How the program is put Together

• Main
  – Creates Canvas – with ArrayList of myMovers
  – Creates ButtonPanel (4 buttons (commands))
    • Creates Button for new TargetFactory
      – When pressed creates new Target
    • Creates Button for new Target2Factory
      – When pressed creates new Target2
    • Creates Button for new CircleFactory
      – When pressed creates new Circle
    • Creates Button for new Circle2Factory
      – When pressed creates new Circle2
    • Target, Target2, Circle, Circle2 put into myMovers when created
  – Creates Slider bar (for target and target2)
Inheritance

- All can use Mover commands....
First Problem - Target

• Modify Target.class
  – One Target object is created
    • This object repeatedly draws the same shape (a circle) each time getting smaller
    • Draws via recursion (recurseDraw method)
    • Which parameter is changing?
      – fillOval must use this change somehow
    • What is the way out?
    • Look at private data - myNumRings
Picture for Target and Target2
Second Problem – Target 2

• Draws same picture – recursion in different place
• Creates multiple objects – one new object with each recursive call
• Constructor – one place with recursion
  – Must create self and create a smaller object with a recursive call to the constructor
• Paint method – one place with recursion
  – Paint current object (fillOval) and then call next smaller object to paint itself if it exists
• Look at private data – one place with recursion
  – Private Target2 myNext;
Target2 Idea – not the picture

Create myself  Start creating next Target2
Target2 Idea – not the picture

Create myself

Start creating
Next Target2
Target2 Idea – not the picture
Another Problem - Circles

• Create only One object - Circles
  – Draws itself and recursively draws four smaller circles (call recurseDraw 4 times)
  – No private state

• Similar approach to Target

• You may want to add parameters to recurseDraw to help you draw the four circles in different places.
Picture for Circles and Circles2

- How to do Colors
- Note in Circle constructor, default color is Black
- You can add parameters to recurseDraw – add one that increments by 1 each time
- Then if that number is even draw one color, if odd then draw another color – pick colors of your choice.
Another Problem: Circles2

• Create multiple objects
• Private data – recursion here
  – Circles2[] myNext;
• Constructor – recursion here
  – Create one object
  – Recursively create an array of size 4 filled with 4 new Circles2
• Paint – recursion here
  – Paint me (the Circle2 object) and then recursively paint its four smaller circles in the array myNext
Circles2 Idea – not the picture

Create me and recursively create 4 smaller Circle2’s
Circles2 Idea – not the picture

For each of those 4 circles they will – create me and recursively
Create 4 smaller circle2’s