ComSci 6
Programming Design and Analysis

February 4, 2010
Prof. Rodger and Prof. Forbes
Announcements

• Reading for next time
  – Chap. 4.6, Chap 7.5, Chap 11.1
  – Reading Quiz due before next class
• Assignment 3 due tonight!
• Assignment 4 out.
Estimation

• Square Root:
  – Given a real number $c$ and some error tolerance $\epsilon$
  – Estimate $t$, the square root of $c$

• Pi:
  – Estimate $\pi$ with a given number of Monte Carlo trials
While Loops: Square Root

• Q. How might we implement `Math.sqrt()`?

• A. To compute the square root of `c`:
  – Initialize `t_0 = c`.
  – Repeat until `t_i = c / t_i`, up to desired precision:
    set `t_{i+1}` to be the average of `t_i` and `c / t_i`.

<table>
<thead>
<tr>
<th><code>t_i</code></th>
<th>=</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>t_0</code></td>
<td>2.0</td>
</tr>
<tr>
<td><code>t_1</code></td>
<td>1.5</td>
</tr>
<tr>
<td><code>t_2</code></td>
<td>1.416666666666665</td>
</tr>
<tr>
<td><code>t_3</code></td>
<td>1.4142156862745097</td>
</tr>
<tr>
<td><code>t_4</code></td>
<td>1.4142135623746899</td>
</tr>
<tr>
<td><code>t_5</code></td>
<td>1.414213562373095</td>
</tr>
</tbody>
</table>

computing the square root of 2

© Sedgewick & Wayne
Newton-Raphson Method

- Square root method explained. \( f(x) = x^2 - c \) to compute \( \sqrt{c} \)
  - Goal: find root of function \( f(x) \).
  - Start with estimate \( t_0 = c \).
  - Draw line tangent to curve at \( x = t_i \).
  - Set \( t_{i+1} \) to be x-coordinate where line hits x-axis.
  - Repeat until desired precision.
Buffon Needle Experiment

Figure 3  The Buffon Needle Experiment
Needle Position

- Needle length = 1, distance between lines = 2
- Generate random $y_{low}$ between 0 and 2
- Generate random angle $\alpha$ between 0 and 180 degrees
- $y_{high} = y_{low} + \sin(\alpha)$
- Hit if $y_{high} \geq 2$

Figure 4
When Does the Needle Fall on a Line?
Constructing objects/Applying methods

- Class Rectangle in Chapter 2
- Creating a Rectangle object with x, y, width, and height
  
  ```java
  Rectangle box = new Rectangle(5, 10, 20, 30);
  ```
- Applying Methods
  
  ```java
  box.translate(15, 25);       // move the rectangle
  System.out.println("x: ", box.getX());   // print x
  System.out.println("y: ", box.getY());   // print y
  ```
Parts of a Class

• State
  – Data
• Constructors
  – Initialize state when object is created
• Accessor methods
  – Accessing data
• Mutator methods
  – Modify data – change the state
Class Example

• Needle class – Needle.java
  – Defines state and behavior of Needle
  – Keeps track of the number of times needle hits the line
  – Use drop() method to simulate dropping needle

• java.util.Random class in Java library
  – nextDouble() generates pseudo-random numbers in [0,1]
import java.util.Random;

/**
 * This class simulates a needle in the Buffon needle experiment.
 */
public class Needle {
  /**
   * Constructs a needle.
   */
  public Needle() {
    hits = 0;
    tries = 0;
    generator = new Random();
  }

  /**
   * Drops the needle on the grid of lines and remembers whether the needle hit a line.
   */
  Continued
22:     public void drop()
23:     {
24:         double ylow = 2 * generator.nextDouble();
25:         double angle = 180 * generator.nextDouble();
26:         // Computes high point of needle
27:         double yhigh = ylow + Math.sin(Math.toRadians(angle));
28:         if (yhigh >= 2) myHits++;
29:         tries++;
30:     }
31:     
32:     /**
33:      * Gets the number of times the needle hit a line.
34:      * @return the hit count
35:      */
36:     public int getHits()
37:     {
38:         return myHits;
39:     }
40:     
41:     Continued

Big Java by Cay Horstmann
Copyright © 2008 by John Wiley & Sons. All rights reserved
/**
 * Gets the total number of times the needle was dropped.
 * @return the try count
 */

public int getTries()
{
    return myTries;
}

private Random myGenerator;
private int myHits;
private int myTries;

Intended Output:

Tries = 10000, Tries / Hits = 3.08928
Tries = 1000000, Tries / Hits = 3.14204
Classwork Today – Loops/Classes

- Snarf the *classwork* project
- Complete Sqrt
  - Finish *estimate* method
  - Print results
- Complete Needle
  - Finish *main* method
  - Print results
- Classwork handout has all the details
- Submit under assignment name *Class07-Feb04*