Lecture 3: ArrayList & Standard Java Graphics

CS 62
Fall 2015
Kim Bruce & Michael Bannister

Lab & Assignment 1

- Strip with 12 squares & 5 silver dollars placed randomly on the board.
- Move silver dollars to fill 5 leftmost squares
  - Coins move only to the left.
  - No coin may pass another.
  - No square may hold more than one coin.
- Last person to move wins.
- Complete description in text.

Read Lab & Assignment Before Lab Wednesday!

Rest of today's lecture is info for lab and assignment!

Random Number Generator

- class Random in java.util package w/ method
  - int nextInt(int n) -- returns random k s.t. 0 ≤ k < n
  - See bottom of pg 30 in text.
- Create Random object once, send nextInt many times.
- See LottoHelper example.

Text Input

- Scanner class
  - Constructor: myScanner = new Scanner(System.in)
    - can use file instead of System.in
    - new Scanner(new File(“filename”))
  - Read values:
    - myScanner.nextInt() -- returns an int
    - myScanner.nextDouble() -- returns a double
    - myScanner.nextLine() -- returns String -- to end of line
    - see documentation for more

Back to assertions in Java

- Won’t use Assert class from Bailey.
- Command to check assertions in standard Java
  - Two forms
    - assert boolExp
    - assert boolExp: message
- Article on when to use assert:
  - http://docs.oracle.com/javase/7/docs/technotes/guides/language/assert.html
  - Short summary -- never use for preconditions of public methods *(OK for private)* — make explicit checks
  - Use for postconditions & class invariants

Turning on assert

- Turn on assertions when run program, by adding “-ea” (without quotes) as virtual machine argument in arguments tab in Eclipse when set up runtime configuration.
- If leave it off, then ignores assert statements.
- If on and assertion is false, then will raise an AssertionError exception and will print associated message

Using Assert & Pre/ postconditions

- Preconditions of public methods must be enforced.
  - But don’t use assert! *Why not?*
- Preconditions of private methods should also be enforced
  - Can use assert to check preconditions of private method
  - Why?
- Use assert to check postconditions and other class invariants

Arrays & ArrayList

Arrays

- Containers that hold objects
  - C[ ] myArray = new C[10]   *// fixed length*
  - Different syntax from objects
  - Public instance variable “length” — *Ugh!*
- Because of limitations of Java virtual machine, cannot create array of type variable:
  - E.g., new T[5] illegal if T is type variable
  - new C[5] is legal if C is primitive, class, or interface name.

ArrayList

- What happens if need more space in array than originally allocated?
- ArrayList is class that creates objects that dynamically expand as needed.
- Part of java.util package
- To get access write import java.util.ArrayList or import java.util.*
- Lab: Squares rep by ArrayList of CoinSquares.

Text uses Vector rather than ArrayList
ArrayList more efficient if no concurrency

ArrayList Specification

- Class ArrayList<E>
- Important methods:
  - add, get, set, indexOf, isEmpty, remove, size, contains, clear
  - size, isEmpty, get, set take constant time
  - add (to end) is “amortized constant” time
- See javadoc at
  - http://download.oracle.com/javase/8/docs/api/

See PostIt example later!

Java Graphics

For details, see document on course web page associated with lecture
Also see GUI cheat sheet in documentation and handouts section.

Overview

- Graphical User Interfaces (GUI)
  - JFrame *(window)*, JPanel *(grouping)*
  - JButton, JTextField, JSlider, JChooser, ...
- Events:
  - Generated by mouse actions, button clicks, etc
  - Use MouseListener, MouseMotionListener, ActionListener, etc. to respond
- Graphics
  - Drawing items on the screen - *today's focus*

Graphics

- Create objects want to draw:
  - Rectangle2D.Double, Line.Double, etc.
  - Constructors take x,y coords and dimensions, but don't actually draw items.
- All drawing takes place in paint method using a "graphics context"
- Triggered implicitly by uncovering window or explicitly by calling repaint method.
  - Adds repaint event to event queue — eventually draws it

Graphics context

- All drawing is done in "paint" method of component
- public void paint(Graphics g)
  - g is a Graphics context provided by system
  - "pen" that does the drawing
  - Programmer calls repaint(), not paint!!
- Need to import classes from java.awt.*, java.awt.geom.*, & javax.swing.*
- See MyGraphicsDemo

General graphic applications

- Create an extension of component (either JPanel, JFrame, or JApplet) and implement paint method in the subclass.
  - See main method of demo to get window to show
  - At start of paint method cast g to Graphics2D to get access to new methods
- Call repaint() on component every time make a change.
  - Causes OS to schedule call of paint in event queue
  - Called automatically if window obscured and revealed

Geometric Objects

- Objects from classes Rectangle2D.Double, Line2D.Double, etc. from java.awt.geom
  - There are also float versions
  - Constructors take params x, y, width, height,
    - but don't draw object
  - myObj.setFrame(x,y,width,height) can move object
  - g2.draw(myObj) -- gives outline
  - g2.fill(myObj) -- gives filled version
  - g2.drawString("a string",x,y) draws string

MyGraphicsDemo

- Class extends JFrame, which creates window.
  - Constructor calls super with title of window.
- Main method creates object, sets size, visibility, and enables go-away box in upper left
- Paint method creates and draws objects.

PostItStdApplication

- More sophisticated
  - JFrame contains two JPanels
  - JFrame uses BorderLayout, so add controls to JPanel in SOUTH, drawing canvas in CENTER of the JFrame
    - Ignore controls for now.
    - See GUI cheat sheet for details
  - DrawingCanvas extends JPanel -- contains paint method
  - Note use of ArrayList to hold PostIts.