Game Engine
Lecture 5 (7/7/2006)

Useless Fact of the Day
Deep Thought Edition

• A Deep Thought, by Jack Handey:
  • Is there anything more beautiful than a beautiful, beautiful flamingo, flying across in front of a beautiful sunset? And he’s carrying a beautiful rose in his beak, and he’s also carrying a very beautiful painting with his feet. And also, you’re drunk.

Topics
• Why it's useful
• Structure
• Important Classes and Methods
• Manual

Why it’s Useful
• Different games have many things in common
  • All draw things (Sprites) on the screen
  • All have a main “game loop,” which runs certain code repeatedly (to move things around, check for collisions, check for user input, etc.)
  • All must be able to start/pause and turn on/off sound
  • The game engine has all the basics already coded in classes that may be reused for every game!
FANG Classes

GameWindow

FrameAdvancer (extends GameWindow)

GameLoop (extends FrameAdvancer)

AnimationCanvas

Sprite

Sprite

Sprite

Tracker

GameLoop

Player

Mouse

Keyboard

Sprite Class

• Visible things on the screen (canvas)
• Has instance variables, mutator and accessor methods for:
  • Shape
  • Location
  • Size
  • Rotation
  • Color
• Three classes: Sprite, StringSprite, ImageSprite

Sprite Class - Simple Example

1. Declare a Sprite object:
   ```java
   Sprite pongPaddle;
   ```

2. Declare and create a Shape object for the sprite:
   ```java
   Rectangle rect;
   rect = new Rectangle(0, 0, 1, 10);
   ```

3. Create the Sprite, giving the shape to the Sprite constructor:
   ```java
   pongPaddle = new Sprite(rect);
   ```

4. Give the Sprite a location, scale, and color (optionally, a rotation [not shown]):
   ```java
   pongPaddle.setLocation(0.0, 0.5);
   pongPaddle.setScale(0.3);
   pongPaddle.setColor(Color.RED);
   ```

5. Add the Sprite to the canvas:
   ```java
   canvas.addSprite(pongPaddle);
   ```

ImageSprites and StringSprites

• An ImageSprite has an image instead of a shape
  • The ImageSprite constructor takes the file name of your image
    • Example:
      ```java
      myImageSprite = new ImageSprite("resources/aPicture.jpg");
      ```

• A StringSprite has words (a string of characters)
  • The StringSprite constructor takes a string
    • Example:
      ```java
      myStringSprite = new StringSprite("blah blah blah blah blah");
      ```
Custom Sprites

- You can make your own custom Sprite class by **extending** the standard Sprite class.
- In the constructor of your new class: (a) Call "super();" (b) Make a shape (or image or string, if you extended ImageSprite or StringSprite), and (c) Call "setShape(yourShape);"
- Example:

  ```java
  public class SquareSprite extends Sprite {
    public SquareSprite() {
      super();
      Rectangle rectangle = new Rectangle(0, 0, 1, 1);
      setShape(rectangle);
    }
  }
  ```

Tracker Class

- Makes Sprites move with some coded-in behavior
- A Tracker object is attached to the Sprite(s) it moves
  - Example: We attach the Tracker object named “ballTracker” to the Sprite named “ball” with “ball.setTracker(ballTracker);”
  - When you write an implementation of the Tracker class, you must define these methods, according to how you want your Tracker to work:
    - Point2D.Double getTranslation()
    - double getScaleFactor()
    - double getRotationAddition()
    - void advanceTime(double timePassed)
  - For an example Tracker class, see ProjectileTracker.java

AnimationCanvas

- Area where all Sprites are drawn
- Can have a background color
- Sprites must be added to the canvas before they will be seen:
  - canvas.addSprite(Sprite theNameOfTheSprite);
- Order of addition equals order the Sprites are drawn on the canvas
- If two Sprites overlap, the Sprite added to the canvas last will appear on top of the other

The Flow of a Game

- 1. “main()” method runs automatically; it starts up the game engine with the “runAsApplication()” method
- 2. Game engine calls “startGame()” method
- 3. When start button is clicked, game engine calls “advanceFrame(double timePassed)” method
  - We repeat this step very quickly, until the end of the game
- 4. If the player clicks the start button again, we repeat steps 2 and 3
Manual

http://www.fangengine.org/doc

- Can also see class components in Eclipse (in the Outline)