

Introduction to Computation and Problem Solving

Class 35: Active Learning: Using Threads to Build an Animation

Prof. Steven R. Lerman
and
Dr. V. Judson Harward

Ticker Application

- Our goal is to use a separate thread to drive an animation that scrolls a string across the screen.
- In the full version, you can stop the animation by pressing the mouse down on the animation screen, and resume the animation by releasing the mouse.
- As a starting point, we give you `Ticker0.java`. Download it from the class web site and let's look at it together. It is not animated. Mouse clicks drive the string across the screen.

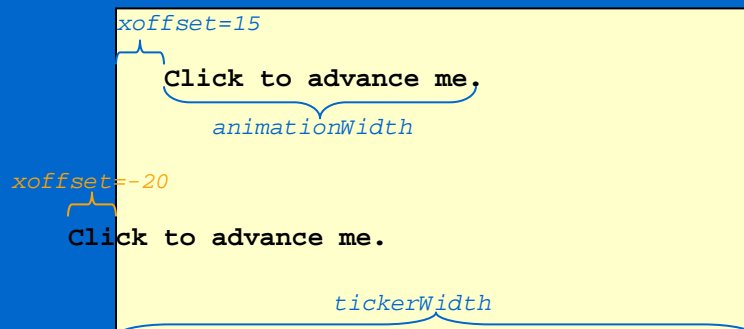
Ticker0, Data Members

```
public class Ticker0 extends JPanel
{
    private String animationString;
    private Font animationFont;
    private int animationWidth;
    private int xoffset;
    private int tickerWidth;

    public static final Dimension DEFAULT_SIZE =
        new Dimension( 400, 200 );
}
```

3

Ticker0, Geometry



4

Ticker0, Constructor

```
public Ticker0(String s) {
    animationString = s;
    setOpaque( true );
    setBackground( Color.black );
    setForeground( Color.white );
    animationFont= new Font("SansSerif",Font.BOLD,16);
    animationWidth = getFontMetrics(animationFont).
        stringWidth( animationString );
    setFont( animationFont );
    xoffset = 15;
    tickerWidth = DEFAULT_SIZE.width;
}
```

5

Ticker0, MouseListener

```
addMouseListener( new MouseAdapter() {
    public void mouseClicked( MouseEvent e )
    {
        // move left 5 pixels each mouse click
        xoffset -= 5;
        // if string off screen, bring it back on right
        if (xoffset <= -animationWidth)
            xoffset = tickerWidth;
        repaint();
    }
} );
} // end constructor
```

6

Ticker0, paintComponent()

```
public void paintComponent(Graphics g)
{
    super.paintComponent( g );
    int yoffset = getHeight() / 2;
    tickerWidth = getWidth(); // allows resize
    g.drawString( animationString, xoffset,
        yoffset );
}
```

7

Ticker, Step 1

- Copy Ticker0 and rename it Ticker1.
- Substitute your favorite movie quote as the Ticker1 constructor argument in the main() method.
- Comment out the MouseListener code for the moment.
- Now modify Ticker1 so that it scrolls continuously, a pixel every 1/100th of a second.
- You may find the model of the SimpleClock program useful.

8

SimpleClock, Constructor

```
public class SimpleClock
  extends javax.swing.JPanel implements Runnable {
  private Font clockFont =
    new Font("SansSerif",Font.BOLD,24);
  private Thread clockThread = null;
  private int xPos = 40;
  private int yPos = 50;

  public SimpleClock() {
    . . .
    clockThread = new Thread( this, "Clock" );
    clockThread.start();
  }
}
```

9

SimpleClock, run()

```
public void run()
{
  while ( true )
  {
    repaint();
    try
    {
      Thread.sleep( 1000 );
    }
    catch (InterruptedException e )
    {}
  }
}
```

10

Ticker, Step 2

- Now, as a second step, copy `Ticker1` and rename it `Ticker2`. Modify `Ticker2` so that the animation stops when you press the mouse and resumes when you release it.
- What events do you have to listen for?
- How will you get the animation to stop and start?