**INSY 4305/5352 Assignment #3**

**Animation using Threads**

**Introduction:**

One of the most common application of using Threads (Runnable interface in particular) is in animating images. In this assignment you will be designing a "simple" moon lander application. (Note that this assignment is more challenging than the previous ones, so start working on it ASAP).

Your application will animate the moon lander ascending and descending. You will be using a customized JPanel class to draw the moon lander image. To implement your JPanel as a thread you need to implement it as a Runnable interface and put the code for the animation in the run method. You will need to make a thread object to launch the thread. Give that thread a reference to the Runnable object in its constructor. The thread then calls the run method of that object, when you call the thread object's start method. This call can be made in the startAnimation method of the JPanel, as shown below:

class ImagePanel extends JPanel implements Runnable

{

private Thread runner;

....

public void startAnimation()

{

if ( runner == null ) {

runner = new Thread(this);

}

runner.start();

}

....

public void stopAnimation()

{

if ( runner != null )

runner = null;

}

....

}

In this case, the this argument to the Thread constructor specifies that the object whose run method should be called when the thread executes is an instance of the ImagePanel object.

Inside the run method of your ImagePanel class you should have a while loop that keeps running as long as the runner thread object is not null, as shown below:

public void run()

{

while ( runner != null )

{

// give your code here

}

}

Using the while loop as shown above provides a very simple mechanism to stop the current thread. All you need to do to stop the current thread is to set the thread variable runner to null anywhere in your JPanel, as shown in the stopAnimation method above. You can restart a new thread by calling the startAnimation method of your JPanel class, which is shown above. This is a common idiom that you will find in many multithreaded applications.

**The Moon Lander Application:**

In this assignment you need to animate moon lander images. There are two images provided below. You will be displaying the lander.jpg image when the lander is descending and displaying the landerFire.jpg image when the lander is ascending. To load the images in your JPanel you need to do the following in its constructor:

Image lander = Toolkit.getDefaultToolkit().getImage( "lander.jpg" );

Image landerFire = Toolkit.getDefaultToolkit().getImage( "landerfire.jpg" );

To display the lander image in your JPanel, do the following in the paintComponent method of your JPanel class:

g.drawImage( lander, x, y, this );

where, x, y are the upper left hand corner coordinates of the JPanel where the image will be displayed. To simulate the ascent or desent of the lander all you need to do is change the y coordinate appropriately inside the run method of your JPanel and call repaint(). Make sure you put the current running thread to sleep for about 0.1 sec. inside the while loop of the run method so that the moon lander's motion is slow and visible.

You can use the getHeight() and getWidth() methods of the JPanel to find out its dimensions so that you can appropriately draw the lander. You can use a boolean variable to track whether the lander is ascending or descending. Once it reaches the top or bottom edge of the JPanel you can toggle the value of that variable. Make sure to draw the lander image when descending and the landerFire image when ascending.

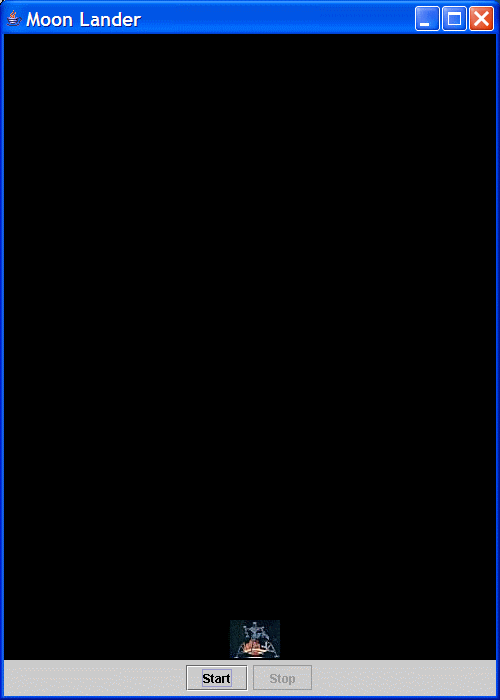
**Deliverables:**

You should submit the appropriate .java files along with the two image files as a single zipped file. You SHOULD name the JPanel class as ImagePanel and your application class as HW3.

[**lander.jpg**](https://elearn.uta.edu/bbcswebdav/pid-4344749-dt-content-rid-15891095_1/users/rsikora/INSY%204305%20my_files/HW/lander.jpg) (Right click on the image to save it)

  
[**landerfire.jpg**](https://elearn.uta.edu/bbcswebdav/pid-4344749-dt-content-rid-15891095_1/users/rsikora/INSY%204305%20my_files/HW/landerfire.jpg)

  
[**Screen shot 1 - At start**](https://elearn.uta.edu/bbcswebdav/pid-4344749-dt-content-rid-15891095_1/users/rsikora/INSY%204305%20my_files/HW/img01.gif)



[**Screen shot 2 - Ascending**](https://elearn.uta.edu/bbcswebdav/pid-4344749-dt-content-rid-15891095_1/users/rsikora/INSY%204305%20my_files/HW/img02.gif)

  
[**Screen shot 3 - Descending**](https://elearn.uta.edu/bbcswebdav/pid-4344749-dt-content-rid-15891095_1/users/rsikora/INSY%204305%20my_files/HW/img03.gif)

