

## About this assignment

This is your first assignment with images. The goal of this assignment is to get you comfortable with accessing the pixels of an image, creating new images, and defining the pixel values of the new image in terms of the original images.

You will be using the classes provided with the book for this assignment. You need be sure that your IDE is setup with the same package for these classes that we went over in class. If you are unsure, you can check by running one of the example programs from my website. Additionally, the filenames and package name for your code is very important for this assignment.

There are three parts, each of which should go into the files `PictureShow.java`, `PictureShrink.java`, and `PictureExpand.java`. They should all be in one package named `assign2_xyz` where `xyz` are your initials (use an `X` if you only have two initials).

Assume that the user select a valid image file when using the `FileChooser.pick()` method. In the case that the file doesn't exist, isn't a valid image file, or can't be loaded for any reason, the behavior of your code can be undefined (in short, don't worry about it).

Note when resizing images, you are expected to perform the operation one pixel at a time, specifically you are not to use an library methods to perform the operation for you.

## Part 1: The Basics

Provide a `FileChooser` and display the selected image. You should also print out the height and width of the image along with the filename that the image came from.

I've basically given you the code for doing this, you're welcome to use it for this part (with proper attribution, of course). This part should serve as a way to test and see that Eclipse is setup properly to use the classes from the book.

The file that implements this program should be called `PictureShow.java`.

## Part 2: Shrinking an Image

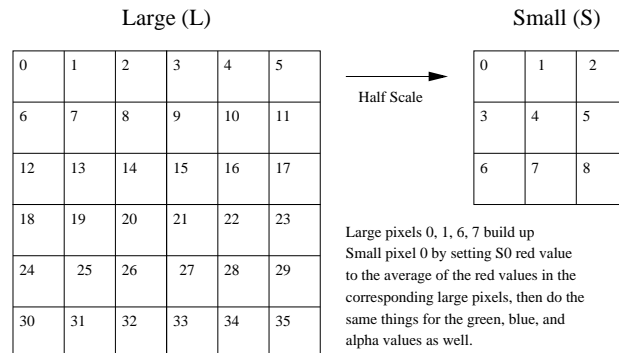
For this program, your task is to load a file (selected using a `FileChooser`), shrink it to half its original size (half the width and half the height), then display it. You should print out the original size of the image and the resulting size. Also display both the original image and the half-size image.

If the image does have dimensions which are both even, you can simply just the image, printing a suitable message the screen. This means that don't have to work about scaling images with even dimensions, but that your program shouldn't crash if one is loaded.

You should use an algorithm that does something more sophisticated than simply discarding half the pixels. The recommended method is to do the following:

The idea is to average the red in four different pixels and put that value as the red in the resulting pixel, then do the same thing for the green, blue, and alpha values as well. When done, you should have a nice looking quarter size picture. Note the book takes a different approach to scaling by simply "sampling" every other pixel of the image, though this is an acceptable approach the result isn't as nice – and is not a valid solution to this problem, though you can definitely draw some ideas from their example.

So basically, the idea is to do something like this:



A few hints on the way:

- There are many ways of doing this problem. You can either use the `getPixels` method to get a long array of all pixels to work on, or you can use the `getPixel` method, which will extract only one pixel at a time, based on the row and column that you pass in to the method. Choose the version that you think works better for you in this case.
- The key to solving this problem is to find the mathematical relationship between the pixel indices in the large scale picture, and pixel indices in the small scale picture.

The file that implements this program should be called `PictureShrink.java`.

## Part 3:

This program will do the opposite of Part 2, it will expand, or enlarge, an image to be twice its original dimensions (that is twice as wide, twice as tall). For this program, your task is to load a file (selected using a `FileChooser`), expand it to twice its original size, then display it. You should print out the original size of the image and the resulting size. Also display both the original image and the double-size image.

The method you should use to do the expansion is bilinear interpolation. Wikipedia has an excellent write up of this technique at [http://en.wikipedia.org/wiki/Bilinear\\_interpolation](http://en.wikipedia.org/wiki/Bilinear_interpolation), and we will also discuss it in class.

Note this technique will not tell you exactly how to handle right-most column and bottom-most line of pixels in your enlarged image. I recommend that you get it working properly, leaving those parts blank (set to black) initially, then deal with them as a special case. You free to use an reasonable technique (we'll talk about a few in class) to deal with these parts of the image, but cannot leave them blank or crop them off. Document in your code what method you are using to handle this special case.

The file that implements this program should be called `PictureExpand.java`.

## Turning it in

You will attach all the three `.java` files that you have created for this this assignment, to the turnin page of WebCT. I Make sure that both your name is clearly documented at the top of each file in an appropriate comment.

Note you will only get one chance to turn in your assignment this time. Be sure that you done with it before handing it in, and that you attach all three files to the assignment.