

Objective: To move and copy pixels in a picture and produce interesting effects.

Background:

Now that we know the basics in modifying pixels, let's try moving them around. If pixels are moved or copied, they can produce interesting effects. Many of these effects we can get from the most basic of graphical editors but they produce amazing results.

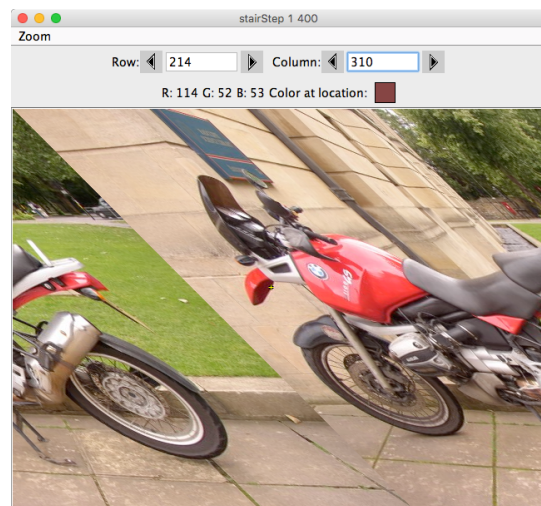
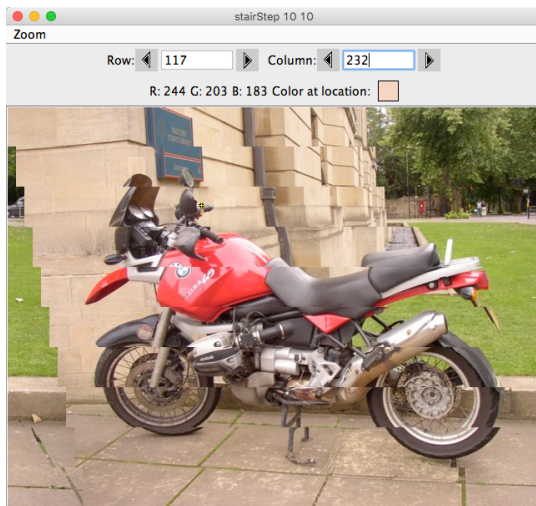
Activities:

A1) **Shifting pixels:** In this activity, we will shift pixels to the right and wrap around to the left. Below are two shifts of the same picture.



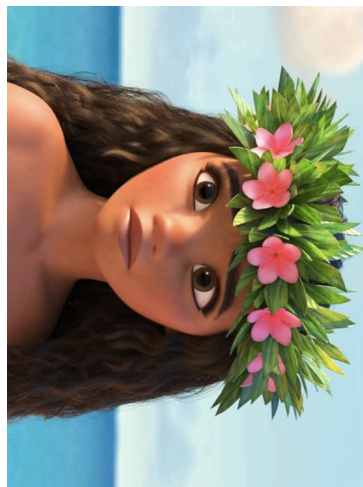
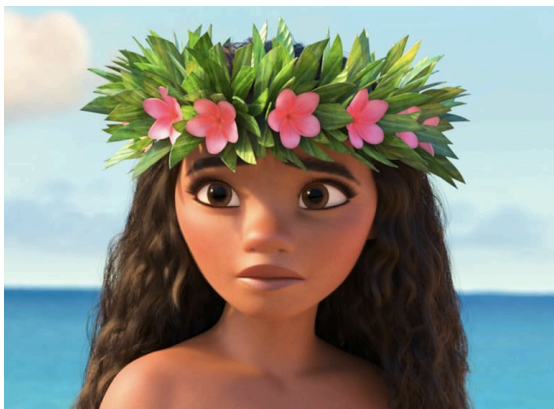
The left photo has been shifted right 25% of the picture. The right photo has been shifted right 50% of the picture. Create a new method `Picture shiftRight(int percent)` inside **Picture.java** to accomplish this effect.

A2) **Stair Step:** A jagged picture can be made using stair steps of shifted pixels. The left picture has 10 stair steps and each step shifts 10 pixels. The right picture has 400 stair steps and shifts each step 1 pixel. Pixels wrap around right to left.



Create a new method `Picture stairStep(int shiftCount, int steps)` inside **Picture.java** to accomplish this effect. **shiftCount** is the number of pixels to shift right at each stair step. **steps** is the number of stair steps. Apply this effect to one of your own pictures.

A3) **Moving pixels:** In this activity, we will rotate the picture 90° clockwise by simply mapping (copying) pixels from the original to the new version. The new picture size should be same as the original but the width and height measures swapped. Create a new method `Picture turn90()` inside **Picture.java** to accomplish this effect. Work the mapping out on paper before you start coding!



A4) **Zoom in:** In this activity, you will zoom into the upper left corner of your image. The zooming effect is achieved by processing only the top-left 25% of the picture area and replicating each pixel to make the image appear larger. More detailed pictures produce better results. Create a new method `Picture zoomUpperLeft()` inside **Picture.java** to accomplish this effect.



A5) **Tiling:** In this activity, you will tile the image. You will reduce the image to 25% of the original area and tile that image in the other three quadrants while mirroring them horizontally and vertically. (See the pictures below). Create a new method `Picture tileMirror()` inside `Picture.java` to accomplish this effect.

