Using the information you have gained so far, modify the code. Click “Submit Assignment” in the upper right corner of the screen to submit your work. Be sure and save the files as YourNameMod13PictureLabAssignmentSeven

1. Write the method mirrorVerticalRightToLeft that mirrors a picture around a mirror placed vertically from right to left. Hint: you can copy the body of mirrorVertical and only change one line in the body of the method to accomplish this. Write a class (static) test method called testMirrorVerticalRightToLeft in PictureTester to test this new method and call it in the main method.
   1. public void mirrorVerticalRightToLeft()
   2. {
   3. Pixel[][] pixels = this.getPixels2D();
   4. Pixel leftPixel = null;
   5. Pixel rightPixel = null;
   6. int width = pixels[0].length;
   7. for (int row = 0; row < pixels.length; row++)
   8. {
   9. for (int col = 0; col < width / 2; col++)
   10. {
   11. leftPixel = pixels[row][width - 1 - col];
   12. rightPixel = pixels[row][col];
   13. rightPixel.setColor(leftPixel.getColor());
   14. }
   15. }
   16. }
2. Write the method mirrorHorizontal that mirrors a picture around a mirror placed horizontally at the middle of the height of the picture. Mirror from top to bottom as shown in the pictures below (Figure 8). Write a class (static) test method in PictureTester to test this new method and call it in the main method.
   1. public void mirrorHorizontal()
   2. {
   3. Pixel[][] pixels = this.getPixels2D();
   4. Pixel leftPixel = null;
   5. Pixel rightPixel = null;
   6. int height = pixels.length;
   7. for (int row = 0; row < height / 2; row++)
   8. {
   9. for (int col = 0; col < pixels[0].length; col++)
   10. {
   11. leftPixel = pixels[row][col];
   12. rightPixel = pixels[height - 1 - row][col];
   13. rightPixel.setColor(leftPixel.getColor());
   14. }
   15. }
   16. }

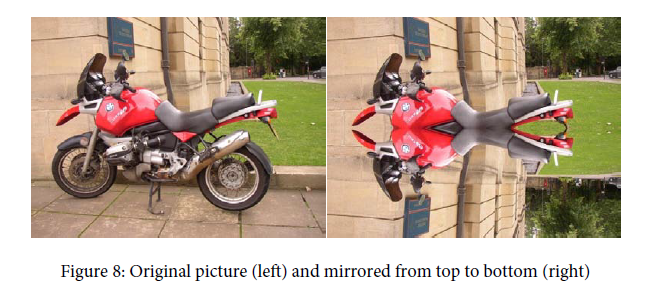


Image from AP College Board Picture Lab Student Guide page 15

1. Write the method mirrorHorizontalBotToTop that mirrors the picture around a mirror placed horizontally from bottom to top. Hint: you can copy the body of mirrorHorizontal and only change one line to accomplish this. Write a class (static) test method in PictureTester to test this new method and call it in the main method.
   1. public void mirrorHorizontalBotToTop() {
   2. Pixel[][] pixels = this.getPixels2D();
   3. Pixel leftPixel = null;
   4. Pixel rightPixel = null;
   5. int height = pixels.length;
   6. for (int row = 0; row < height / 2; row++)
   7. {
   8. for (int col = 0; col < pixels[0].length; col++)
   9. {
   10. leftPixel = pixels[height - 1 - row][col];
   11. rightPixel = pixels[row][col];
   12. rightPixel.setColor(leftPixel.getColor());
   13. }
   14. }
   15. }
2. Extra Credit — Work in groups to figure out the algorithm for the method mirrorDiagonal that mirrors just a square part of the picture from bottom left to top right around a mirror placed on the diagonal line (the diagonal line is the one where the row index equals the column index). This will copy the triangular area to the left and below the diagonal line as shown below. This is like folding a square piece of paper from the bottom left to the top right, painting just the bottom left triangle and then (while the paint is still wet) folding the paper up to the top right again. The paint would be copied from the bottom left to the top right as shown in the pictures below (Figure 9). Write a class (static) test method in PictureTester to test this new method and call it in the main method.
   1. // mirrors from top right to bottom left
   2. public void mirrorDiagonal()
   3. {
   4. Pixel[][] pixels = this.getPixels2D();
   5. Pixel topRightPixel = null;
   6. Pixel bottomLeftPixel = null;
   7. int maxLength;
   8. // ensure that no NullPointerExceptions occur
   9. // if row < column, max = column
   10. if (pixels.length < pixels[0].length) {
   11. maxLength = pixels.length;
   12. }
   13. // if column < row, max = row
   14. else {
   15. maxLength = pixels[0].length;
   16. }
   18. for (int row = 0; row < maxLength; row++)
   19. {
   20. for (int col = row; col < maxLength; col++)
   21. {
   22. // store top right pixel
   23. topRightPixel = pixels[row][col];
   24. // store bottom left pixel
   25. bottomLeftPixel = pixels[col][row];
   26. // swap at bottom left
   27. bottomLeftPixel.setColor(topRightPixel.getColor());
   28. }
   29. }
   30. }

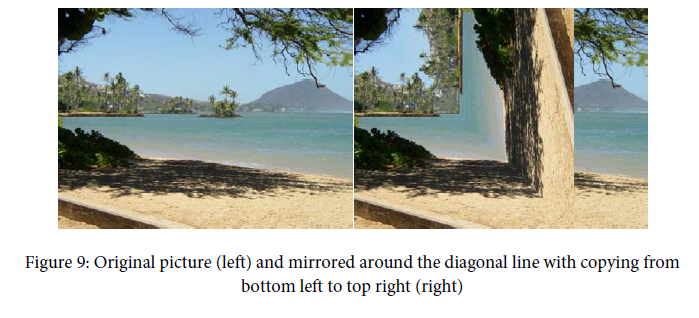


Image from AP College Board Picture Lab Student Guide page 16