**Picture Lab – Activity 7: Mirroring part of a picture**

**Questions**

1. How many times would the body of this nested for loop execute? \_\_\_90\_\_\_\_\_\_\_\_\_\_\_\_

for (int row = 7; row < 17; row++)

for (int col = 6; col < 15; col++)

2. How many times would the body of this nested for loop execute? \_105\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

for (int row = 5; row <= 11; row++)

for (int col = 3; col <= 18; col++)

**Exercises -** Copy and paste the methods from the Picture class below once you have them working.

• Check the calculation of the number of times the body of the nested loop executes by adding an integer count variable to the mirrorTemple method that starts out at 0 and increments inside the body of the loop. Copy and Paste the mirrorTemple method below. Print the value of count after the nested loop ends.

public void mirrorTemple()

{

int mirrorPoint = 276;

Pixel leftPixel = null;

Pixel rightPixel = null;

int count = 0;

Pixel[][] pixels = this.getPixels2D();

// loop through the rows

for (int row = 27; row < 97; row++)

{

// loop from 13 to just before the mirror point

for (int col = 13; col < mirrorPoint; col++)

{

leftPixel = pixels[row][col];

rightPixel = pixels[row]

[mirrorPoint - col + mirrorPoint];

rightPixel.setColor(leftPixel.getColor());

count++;

}

}

System.out.println(count);

}

18410

• Write the method mirrorArms to mirror the arms on the snowman (“snowman.jpg”) to make a snowman with 4 arms. Write a class (static) test method in PictureTester to test this new method and call it in the main method.

public void mirrorArms(){

Pixel leftPixel = null;

Pixel rightPixel = null;

Pixel[][] pixels = this.getPixels2D();

// loop through the rows

for (int row = 157; row < 190; row++)

{

// loop from 13 to just before the mirror point

for (int col = 100; col < 170; col++)

{

leftPixel = pixels[row][col];

rightPixel = pixels[190 - row + 190][col];

rightPixel.setColor(leftPixel.getColor());

}

}

for (int row = 171; row < 205; row++)

{

// loop from 13 to just before the mirror point

for (int col = 240; col < 295; col++)

{

leftPixel = pixels[row][col];

rightPixel = pixels[190 - row + 190][col];

rightPixel.setColor(leftPixel.getColor());

}

}

}

• Write the method mirrorGull to mirror the seagull (“seagull.jpg”) to the right so that there are two seagulls on the beach near each other. Write a class (static) test method in PictureTester to test this new method and call it in the main method.

public void mirrorGull(){

Pixel leftPixel = null;

Pixel rightPixel = null;

Pixel[][] pixels = this.getPixels2D();

for (int row = 227; row < 321; row++)

{

// loop from 13 to just before the mirror point

for (int col = 235; col < 345; col++)

{

leftPixel = pixels[row][col];

rightPixel = pixels[row][345 - col + 345];

rightPixel.setColor(leftPixel.getColor());

}

}

}