**Pseudocode**

1. Create a formula to draw one rectangle, and name it ‘makeRect’.
2. Have the necessary parameters include:
   1. The turtle object being used.
      1. This is the turtle object created, ‘myrtle’.
   2. The color of the turtle and line.
      1. Save this as ‘color’.
   3. The x and y positions of the starting position.
      1. Save these as integers ‘startx’ and ‘starty’, respectively.
   4. The x and y positions of the ending position.
      1. Save these as integers ‘endx’ and ‘endy’, respectively.
   5. The thickness of the line that will be drawn.
      1. Save this as an integer named ‘thickness’.
3. Hide the turtle object.
4. Raise the pen.
5. Set the color to the variable ‘color’.
6. Set the pen width to the variable ‘thickness’.
7. Move the turtle to the x value defined in ‘startx’ and the y value defined in ‘starty’.
8. Lower the pen.
9. Move the turtle to the x value defined in ‘endx and the y value defined in ‘starty’.
   1. *(Line 1, horizontal line from beginning/top left point to top right point).*
10. Move the turtle to the x value defined in ‘endx and the y value defined in ‘endy’.
    1. *(Line 2, vertical line from top right point to bottom right point).*
11. Move the turtle to the x value defined in ‘startx’ and the y value defined in ‘endy’.
    1. *(Line 3, horizontal line from bottom right point to bottom left point).*
12. Move the turtle to the x value defined in ‘startx’ and the y value defined in ‘starty’.
    1. *(Line 4, vertical line from bottom left point to beginning/top left point).*
13. Create a new formula to wait for a defined amount of time, and name it ‘wait’.
14. Have the necessary parameter be ‘ms’, which is time in milliseconds.
15. Import code to wait for ‘ms’ amount of time.
16. Create a canvas for the turtle to draw on
17. Create a new turtle object, named myrtle.
18. Create the following variables. (They can be set to any value since it will change with the user’s choice).
    1. Integer userInput, set it to 0.
    2. Integer n, set it to 0.
    3. Integer speed, set it to 0.
    4. Integer thick1, set it to 0.
    5. Integer thick2, set it to 0.
19. Ask the user how many times they would like to replay the animation.
20. Save it in the integer named userInput.
21. Ask the user what speed they want the animation to run at, in milliseconds.
22. Save it in the integer named speed.
23. Ask the user what thickness do they want the first rectangle to be.
24. Save it in the integer named thick1.
25. Ask the user what thickness do they want the second rectangle to be.
26. Save it in the integer named thick2.
27. For the animation, create a do while loop.
28. First, create the green rectangle by using the correct parameters of the ‘makeRect’ formula created earlier.
    1. Set the turtle to myrtle .
    2. Set the color to green.
    3. Set ‘startx’ to 60.
    4. Set ‘starty’ to 80.
    5. Set ‘endx’ to 240.
    6. Set ‘endy’ to 160.
    7. Set thickness to ‘thick1’.
29. Next, wait for ‘speed’ amount of seconds.
30. Next, create the white box that hides the first rectangle by using the ‘makeRect’ formula.
    1. Set the turtle to myrtle .
    2. Set the color to white.
    3. Set ‘startx’ to 0.
    4. Set ‘starty’ to 0.
    5. Set ‘endx’ to 300.
    6. Set ‘endy’ to 300.
    7. Set thickness to 600.
31. Next, create the second pink rectangle by using the correct parameters of the ‘makeRect’ formula.
    1. Set the turtle to myrtle .
    2. Set the color to pink.
    3. Set ‘startx’ to 30.
    4. Set ‘starty’ to 20.
    5. Set ‘endx’ to 280.
    6. Set ‘endy’ to 300.
    7. Set thickness to ‘thick2’.
32. Next, wait for the ‘speed’ amount of seconds.
33. Next, create the white box that hides the first rectangle by using the ‘makeRect’ formula.
    1. Set the turtle to myrtle .
    2. Set the color to white.
    3. Set ‘startx’ to 0.
    4. Set ‘starty’ to 0.
    5. Set ‘endx’ to 300.
    6. Set ‘endy’ to 300.
    7. Set thickness to 600.
34. Add one to the counter ‘n’.
35. The while condition of the loop is while n is less than the ‘userInput’, so the user can control how many time the animation repeats.