Samuel Chong

2/12/19

CS 2302

Lab 1

MW 10:30-11:50

Olac Fuentes

Introduction:

The problem that I tried to solve for this lab was trying to create figures and make sub-figures by calling the method with recursion. The first figure was a square with sub-squares in each corner. The second figure was a circle and the sub-circles must be coming from the left of the circle, not the center. The third one was a binary tree. Lastly, the last figure was a circle that had 5 sub-circles inside of it, and inside the small circles, they had even smaller circles inside of them.

Proposed solution design and implementation:

To try and figure out how to solve these problems, I started playing around with the codes provided by the professor (circle and square). After doing some tests, I noticed that in the circle code, if you add radius to x or y, the sub-circles moved positions. So, when I added radius to x in the line where it draws the figure, I accidentally managed to solve a problem the lab required us to do. That was one figure I had completed. The hard part was trying to get the correct number of circles each figure needed to get the same figure as the image. So, I just kept guessing until getting correct. The code provided by the professor was the key in order to make the figures

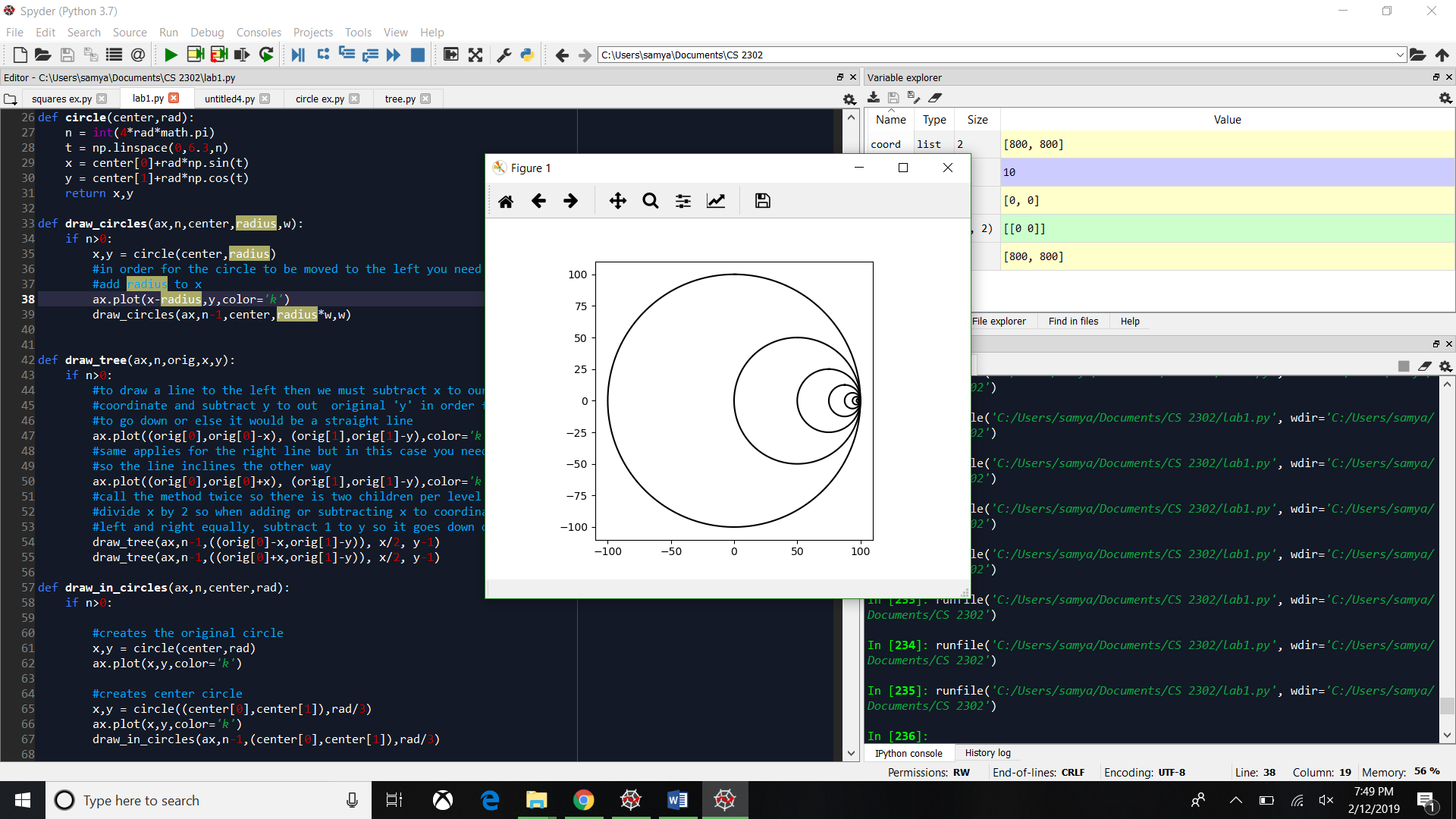
After doing the circles I started to do the squares figure because of the code provided by the professor. After learning how the code work, I noticed the line “p = np.array” is where you plug in the coordinates in order to create a complete square. If you were missing a coordinate, then it would not create a complete circle or even it could create a triangle instead. In order to create a square, we needed different coordinates so it would make the square, so I subtracted and added the radius in different order to create the square figure, the first and last had to be the same or it would not create a square. After having the coordinates, I made a recursive call but noticed it only made one, so I realized that I need 4 recursive calls, each of them different by, again, adding and subtracting the radius. The professor told us the sub-squares depended on r so that is where I got the idea that I needed to add and subtract the radius.

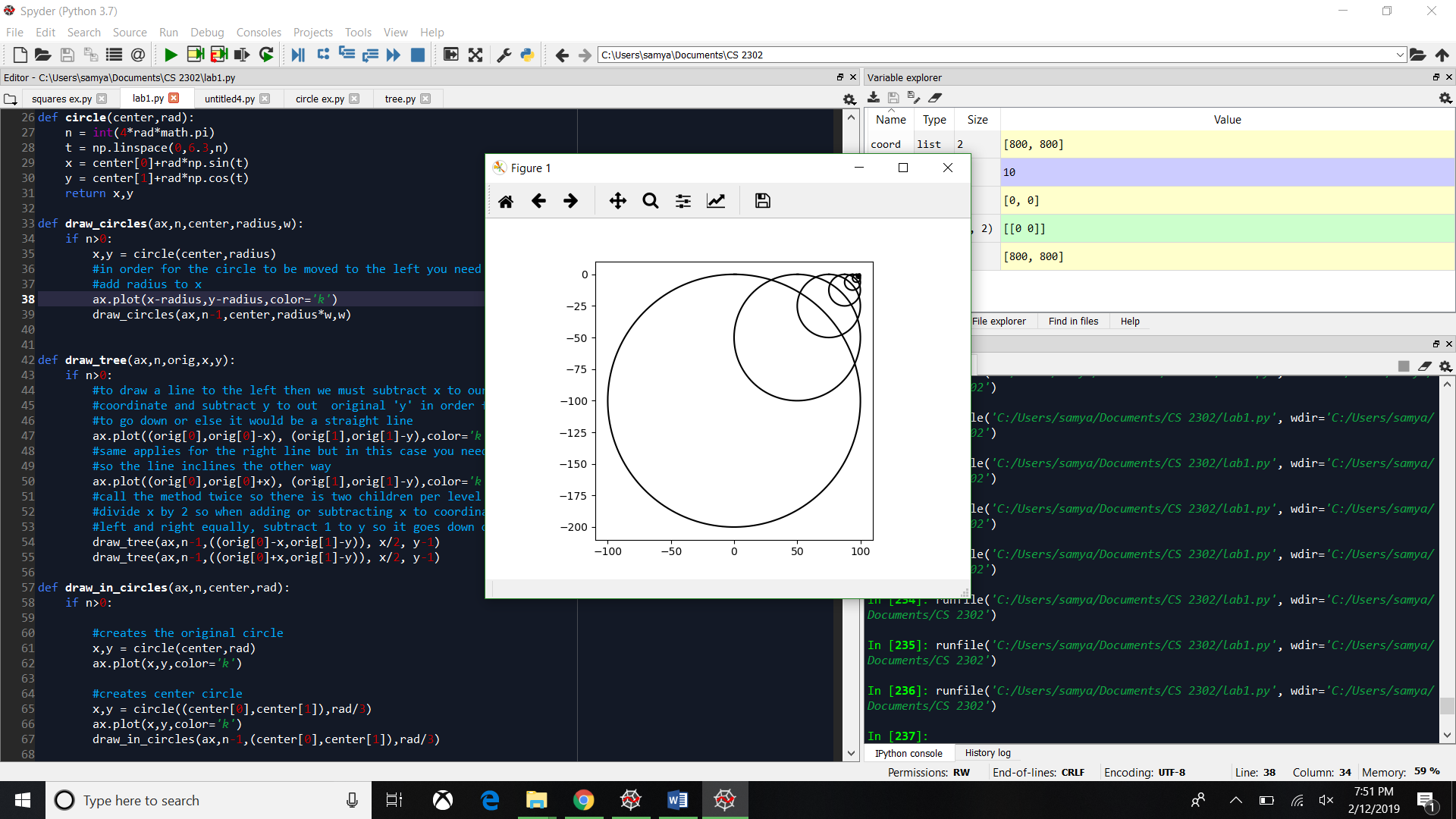
For the binary tree it was a hard one because it was the first code, we had to make without an example provided by the professor. I knew it was somehow related to the square one because the binary tree is just drawing straight lines, same as the square. So, I then noticed we needed to use coordinates to draw the lines, but I kept getting straight lines and not tilted like the binary tree. I couldn’t find the answer to it until the professor told us in class, we needed to subtract y in the coordinates in order it goes down one level, and to make the lines go left and right we needed to add and subtract x to the coordinates. The same applied to the binary tree as the squares, it needed more then one recursion because it only kept making one line instead of the both sides.

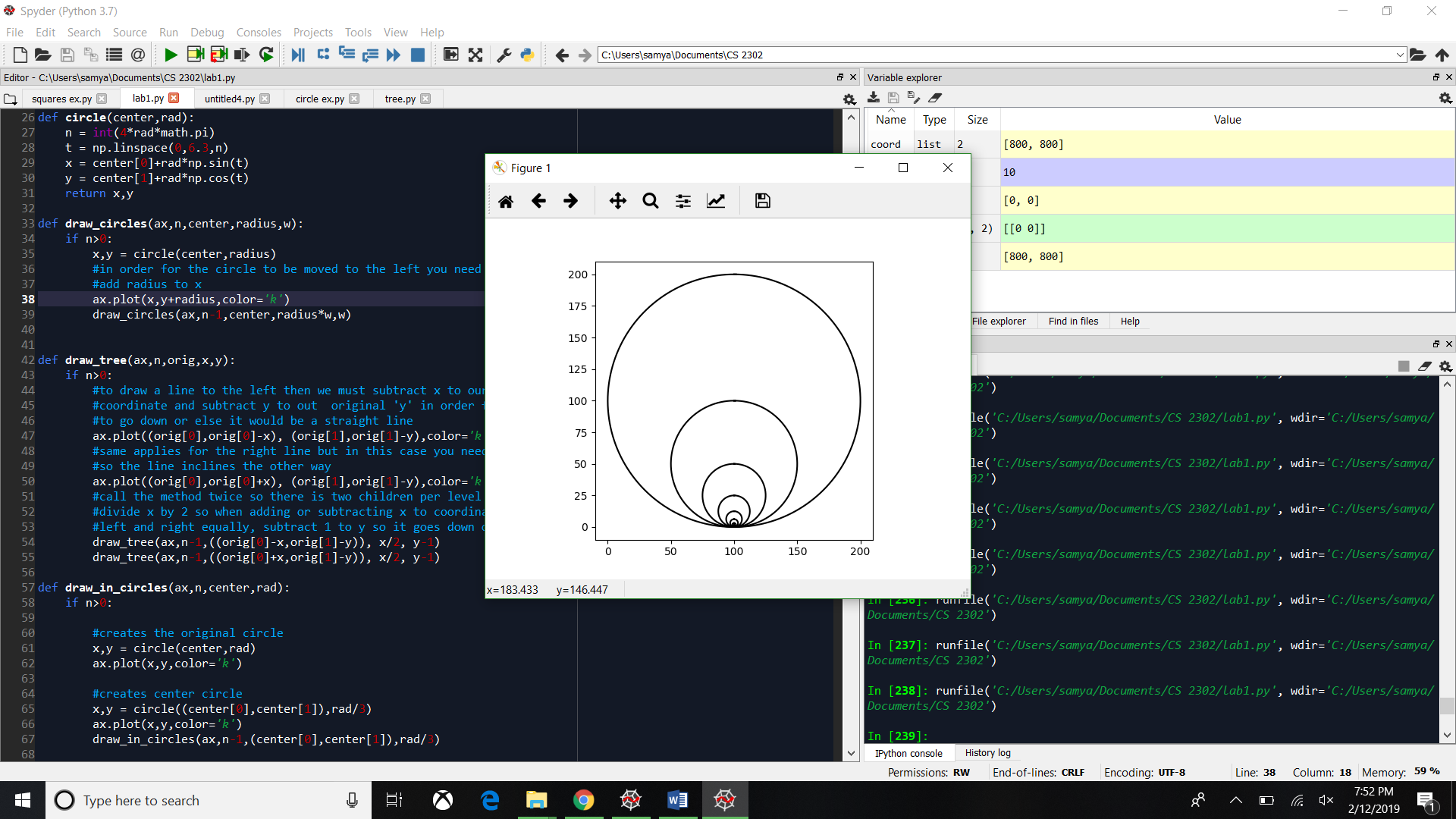
The 5 circles was toughest of all. I could find a way solve it. I approached the same as the other circle figure. I started added and subtracting the radius to x and y, and noticed the circles were moving into place, but when I had for example x+radius in one line and x-radius in the next line, it would create very strange figures. Instead of making sub-circles inside the circle, it would make circles the size of the original but next to the original circle. I was not able to find a solution in time, and I also needed help, so I went to the Tas office hours for help and she helped me with my problem. I needed to use coordinates when calling the circle method. My circles were not the right size as well and in the office hours I learned the radius needed to be divided by 3 instead of 2. When I kept working on my problems, I noticed my inner circles where not in the right place, there was no space for the center circle, so I knew something with the radius was off. I started moving the division and so on and I was able to position them correctly by multiplying by 2 the radius divided by 3, that got the circles in order.

Experiments:

Circle experiments

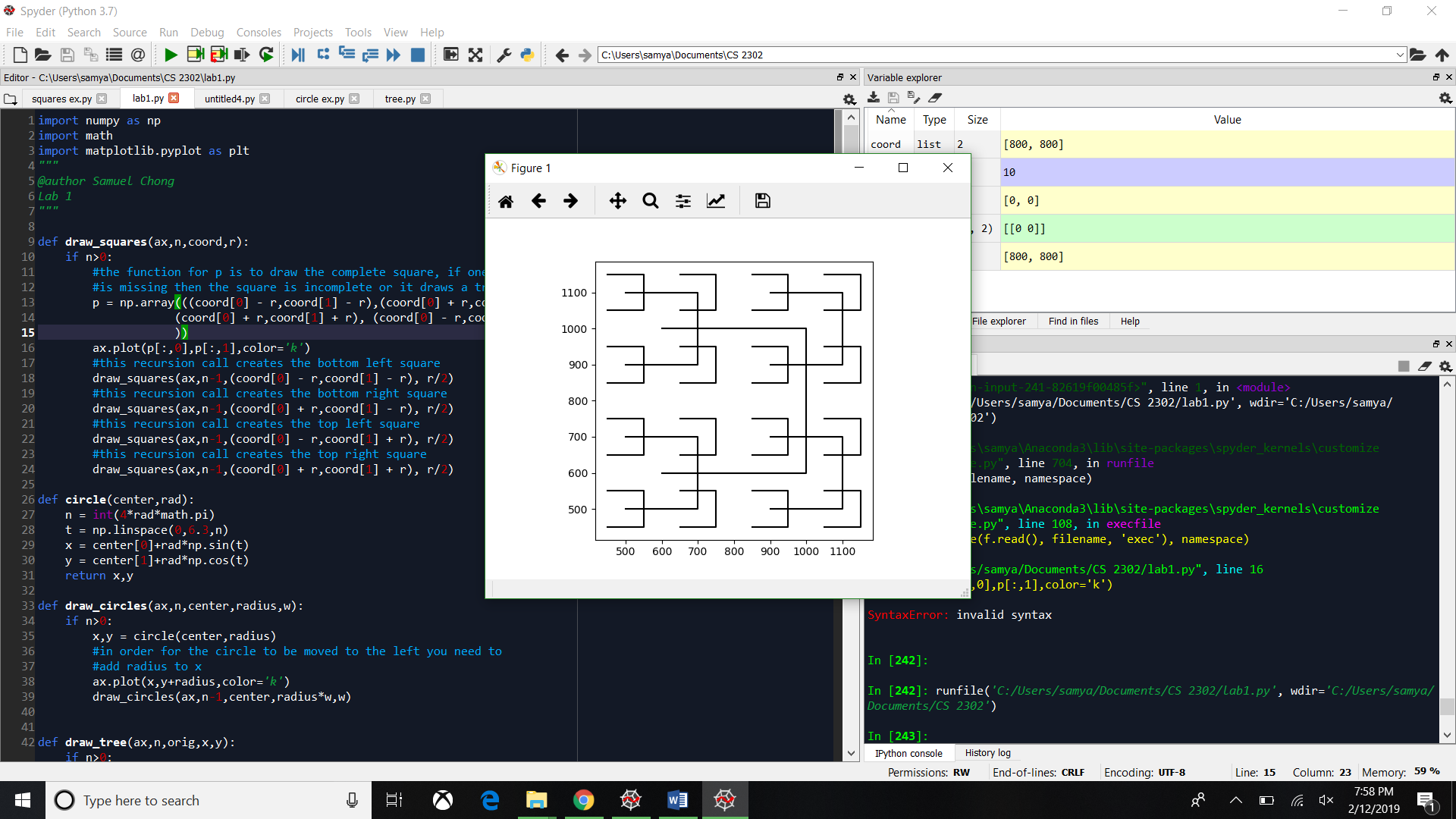
When subtracting radius to x

When subtracting radius to x and y When adding radius to y

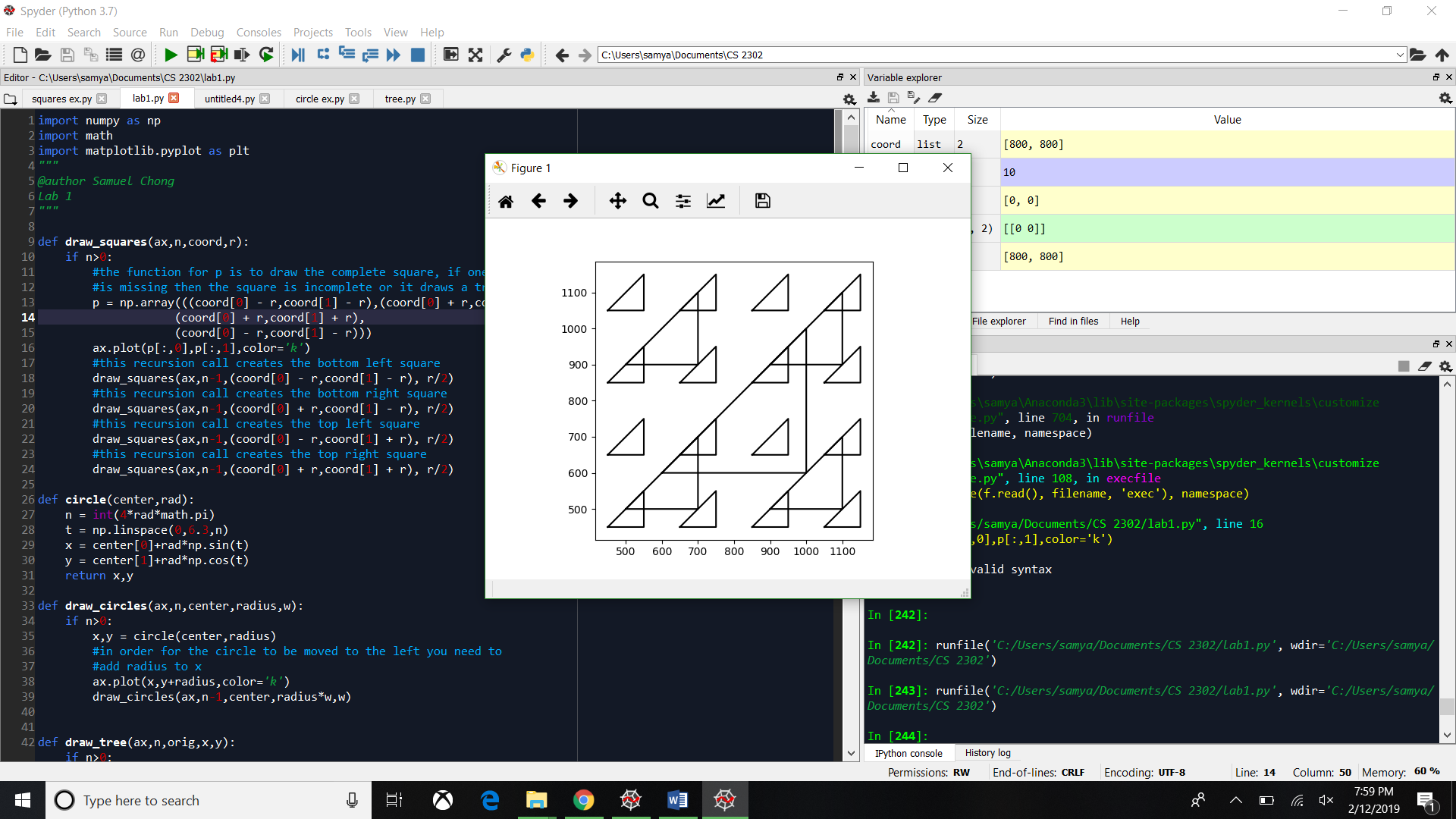


Square experiments

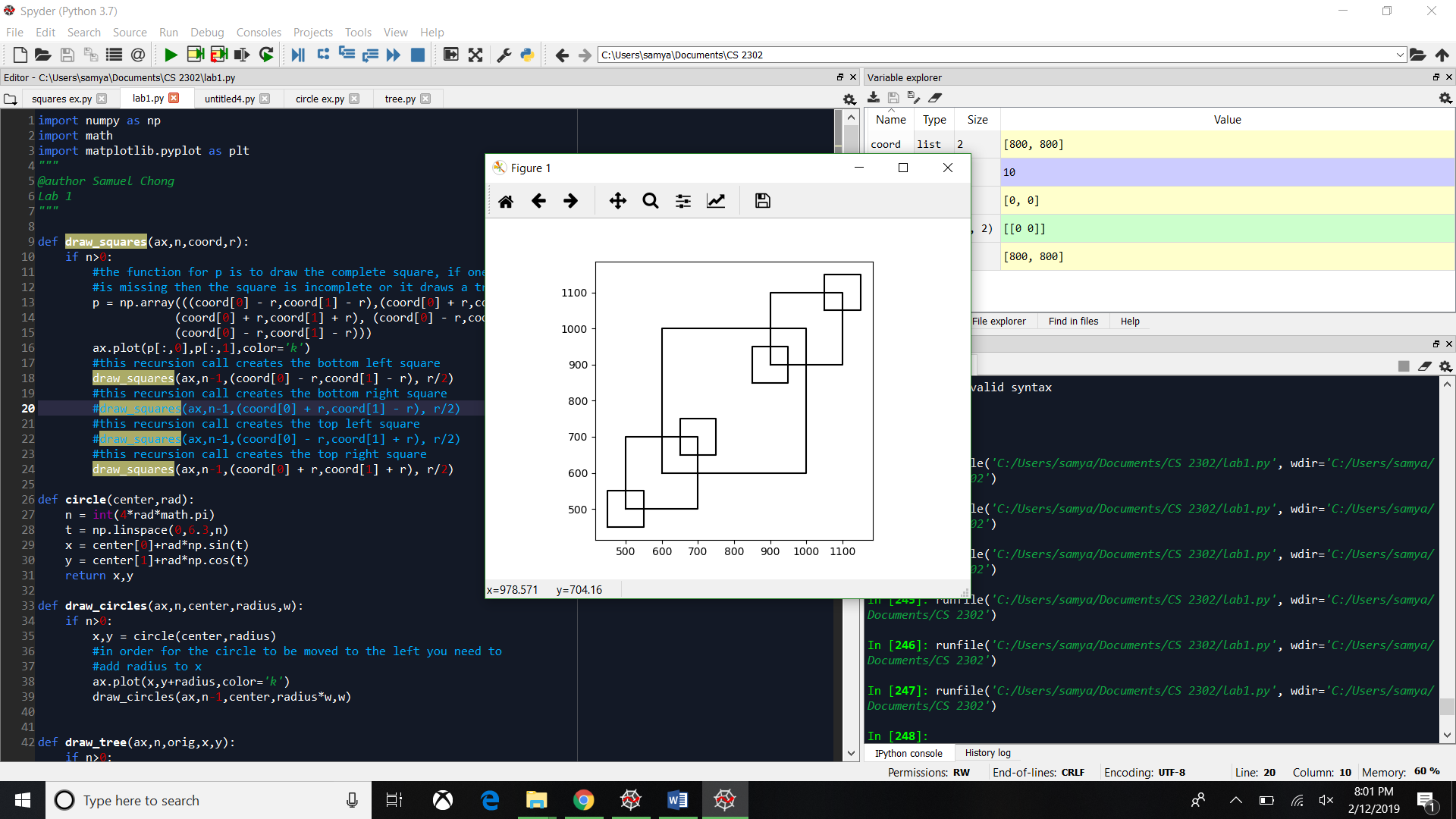
Not having the 5th coordinate that completes the complete square



Missing one coordinate that is not the first or last will cause a triangle



When only having 2 recursive calls instead of 4

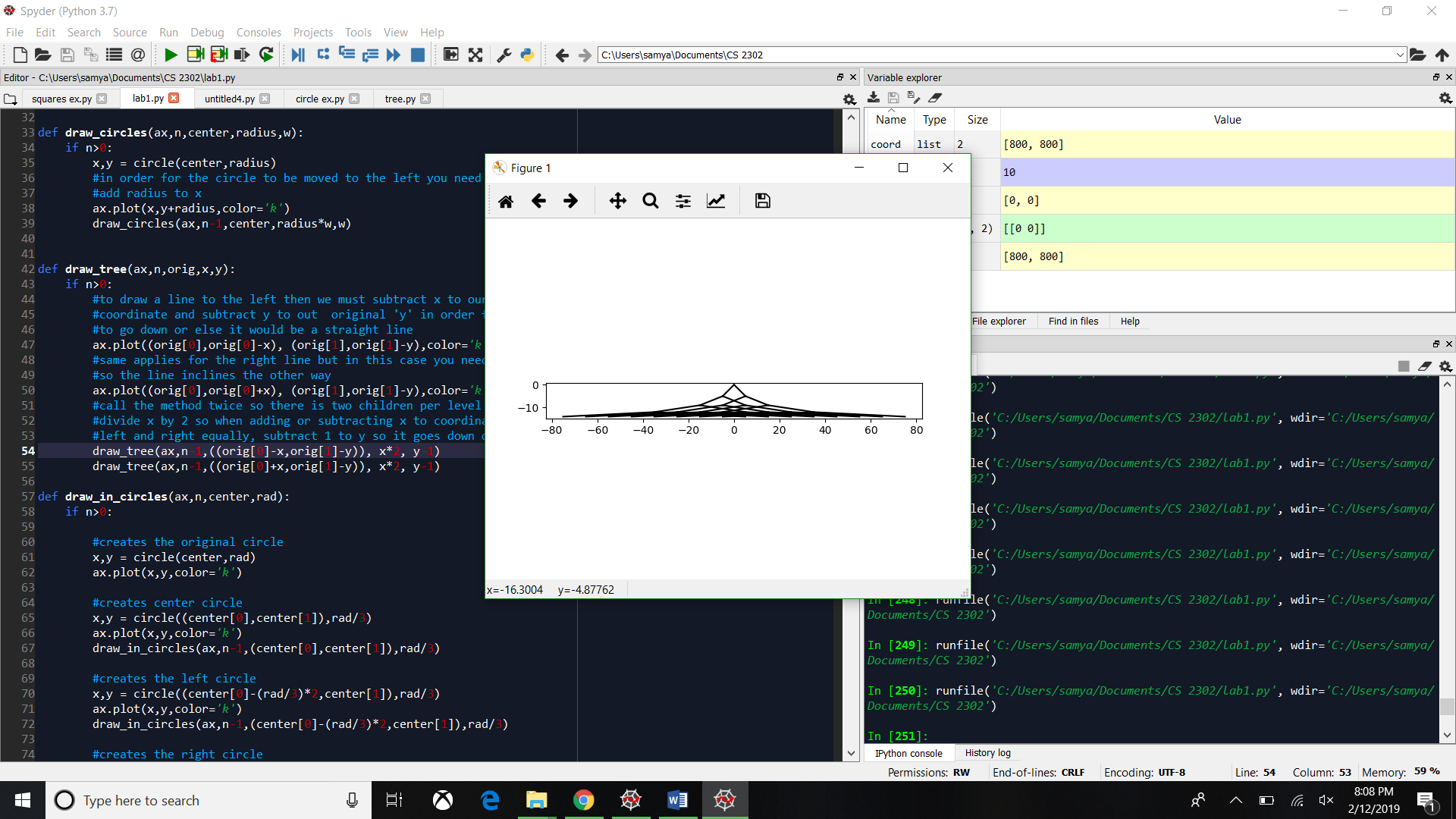


Binary tree experiments

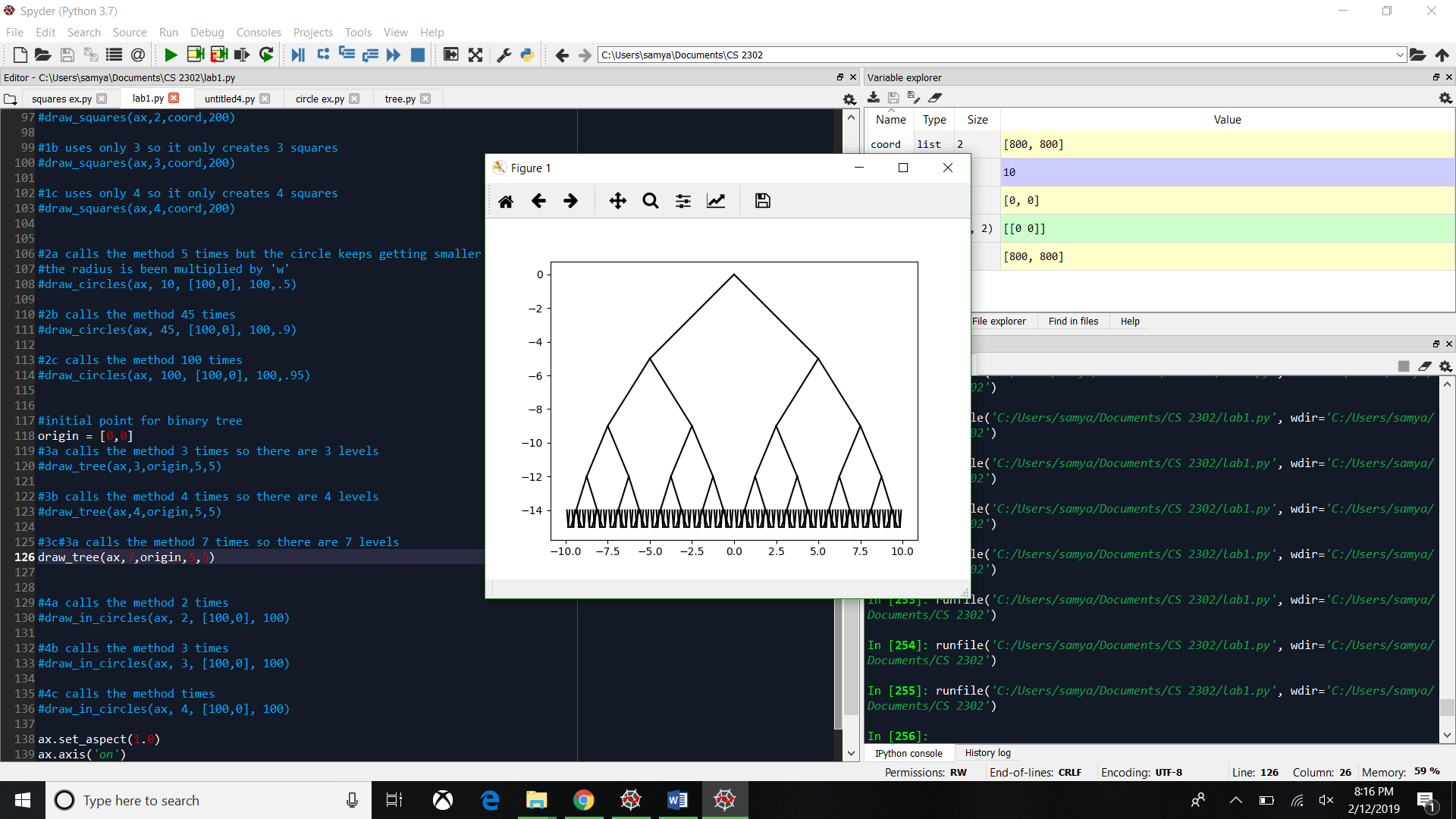
When I didn’t add or subtract x to the coordinates



When making the recursion call, I multiplied x times 2 instead of diving it

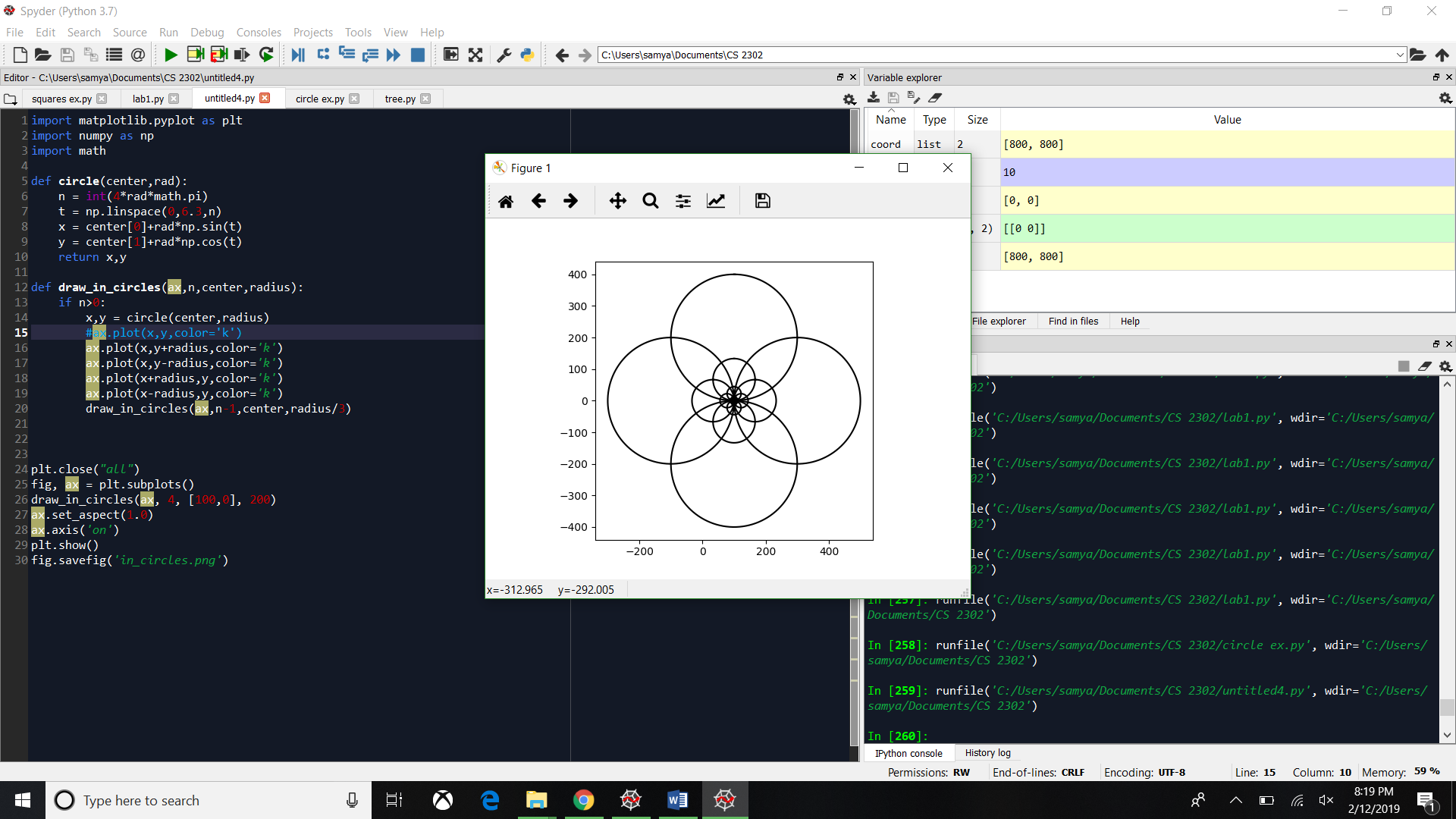


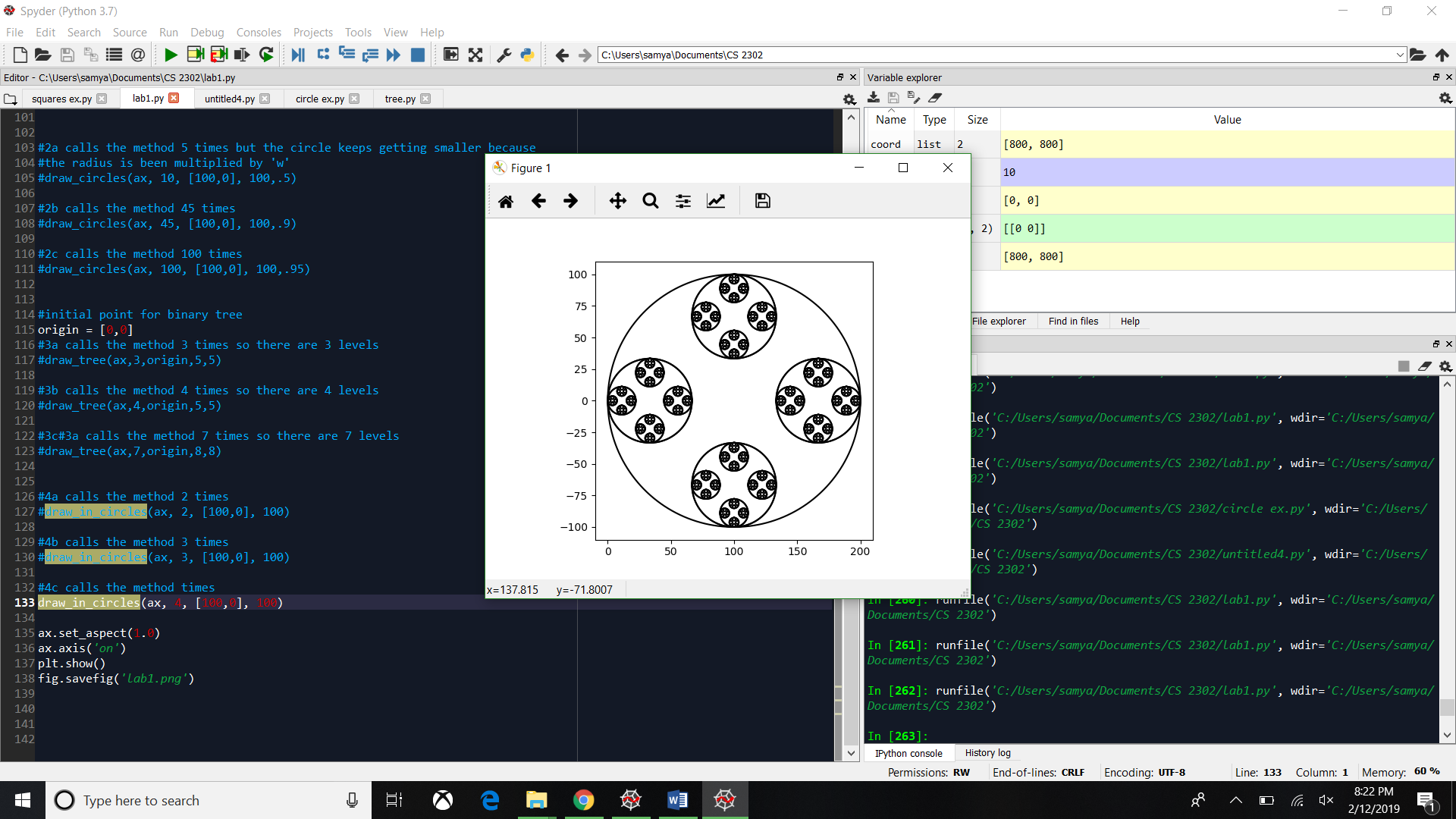
When I call the method 7 times and x and y are 5



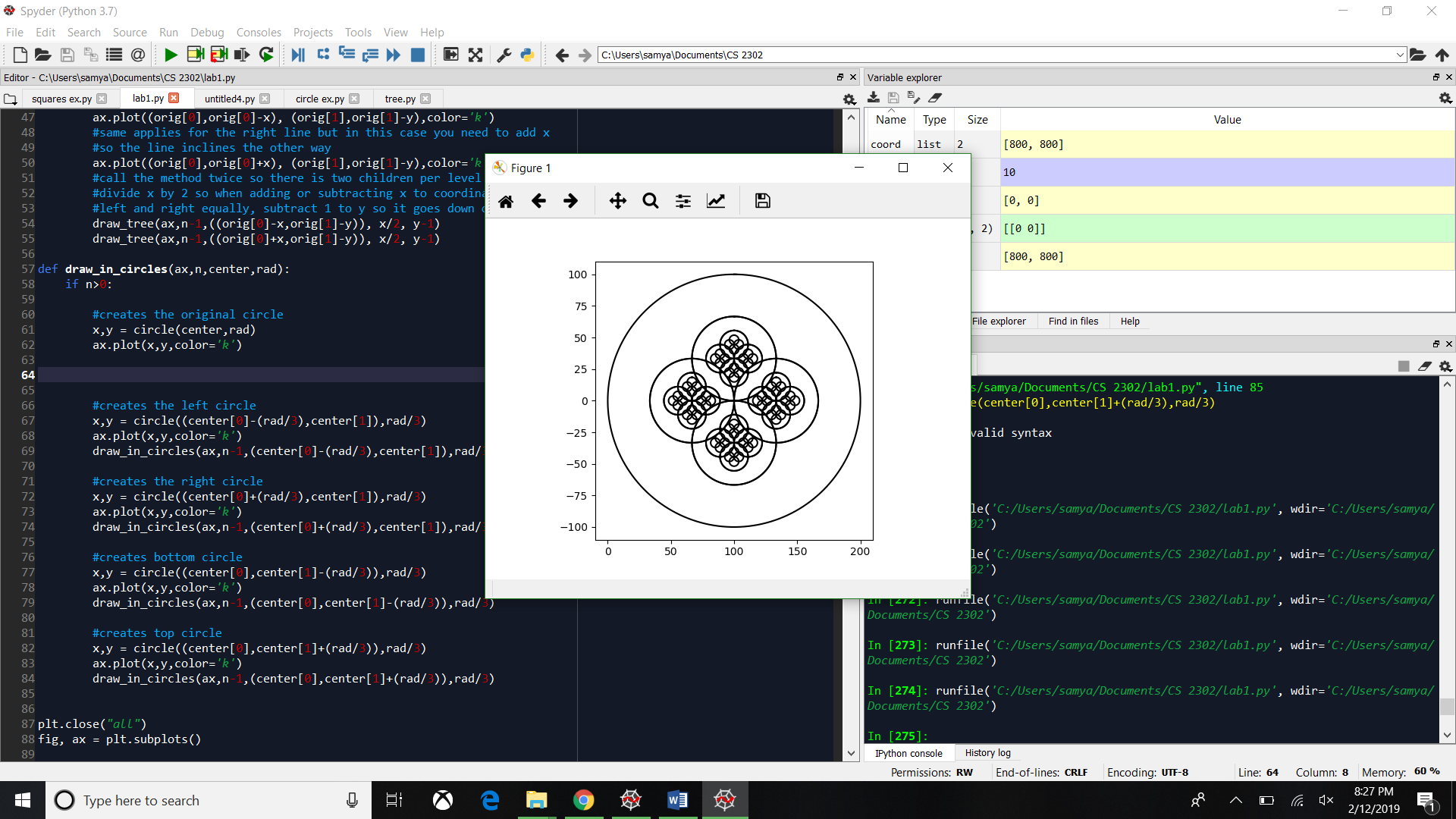
5 Circles experiments

When adding and subtracting radius to x and y instead of being a coordinate

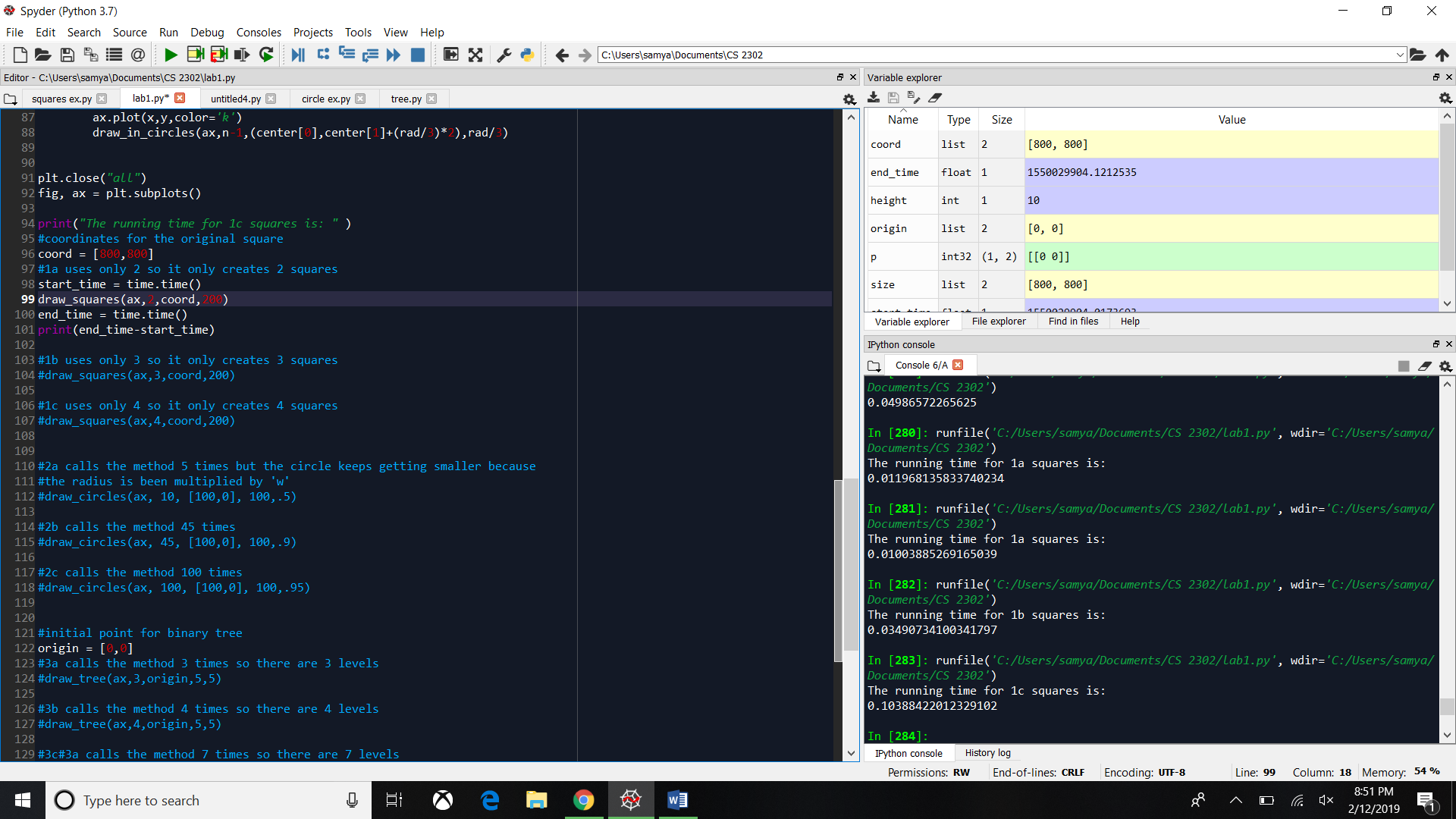


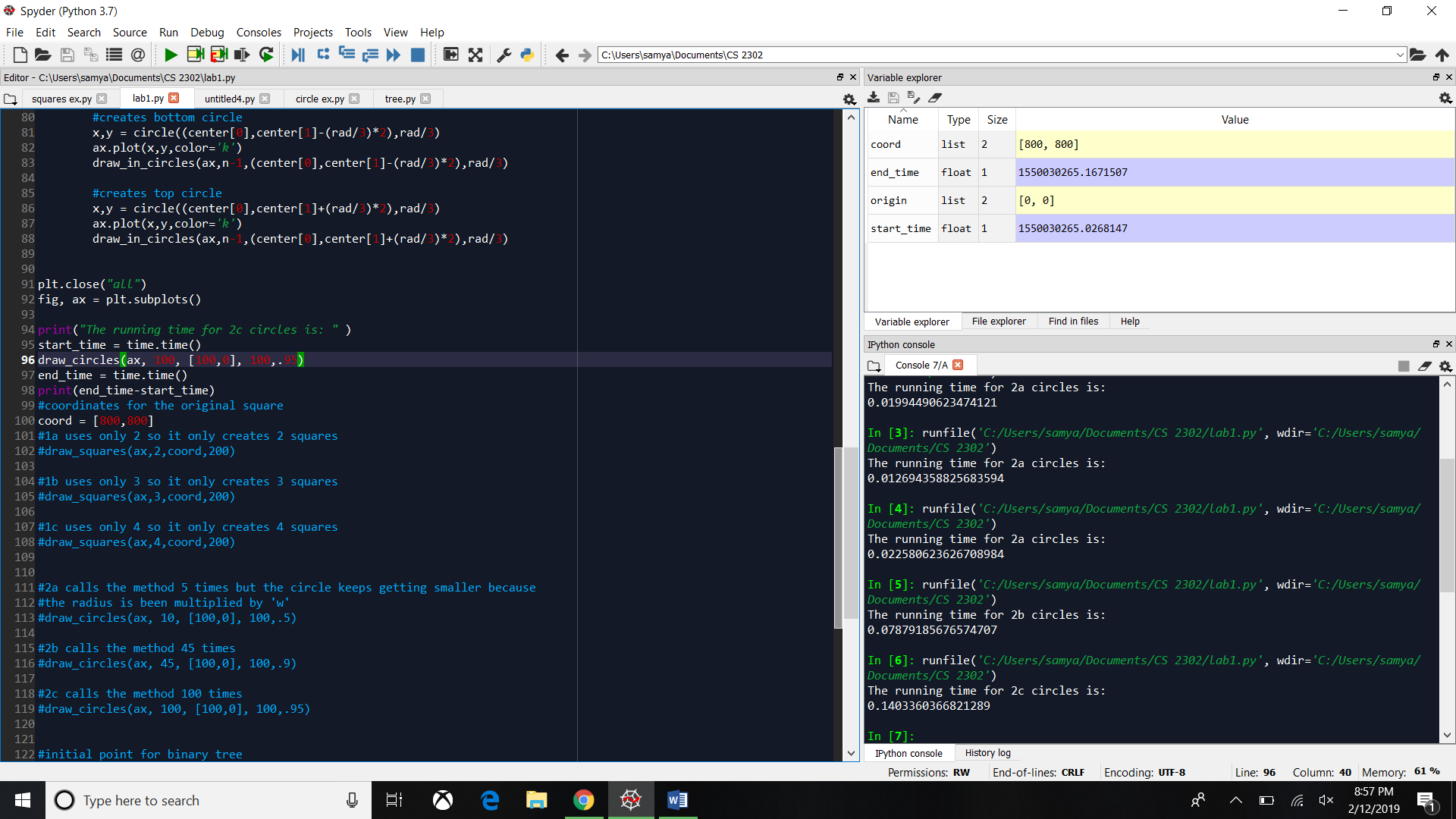
Not having the center circle

Not having the center circle and not multiplying times 2 the radius divided by 3

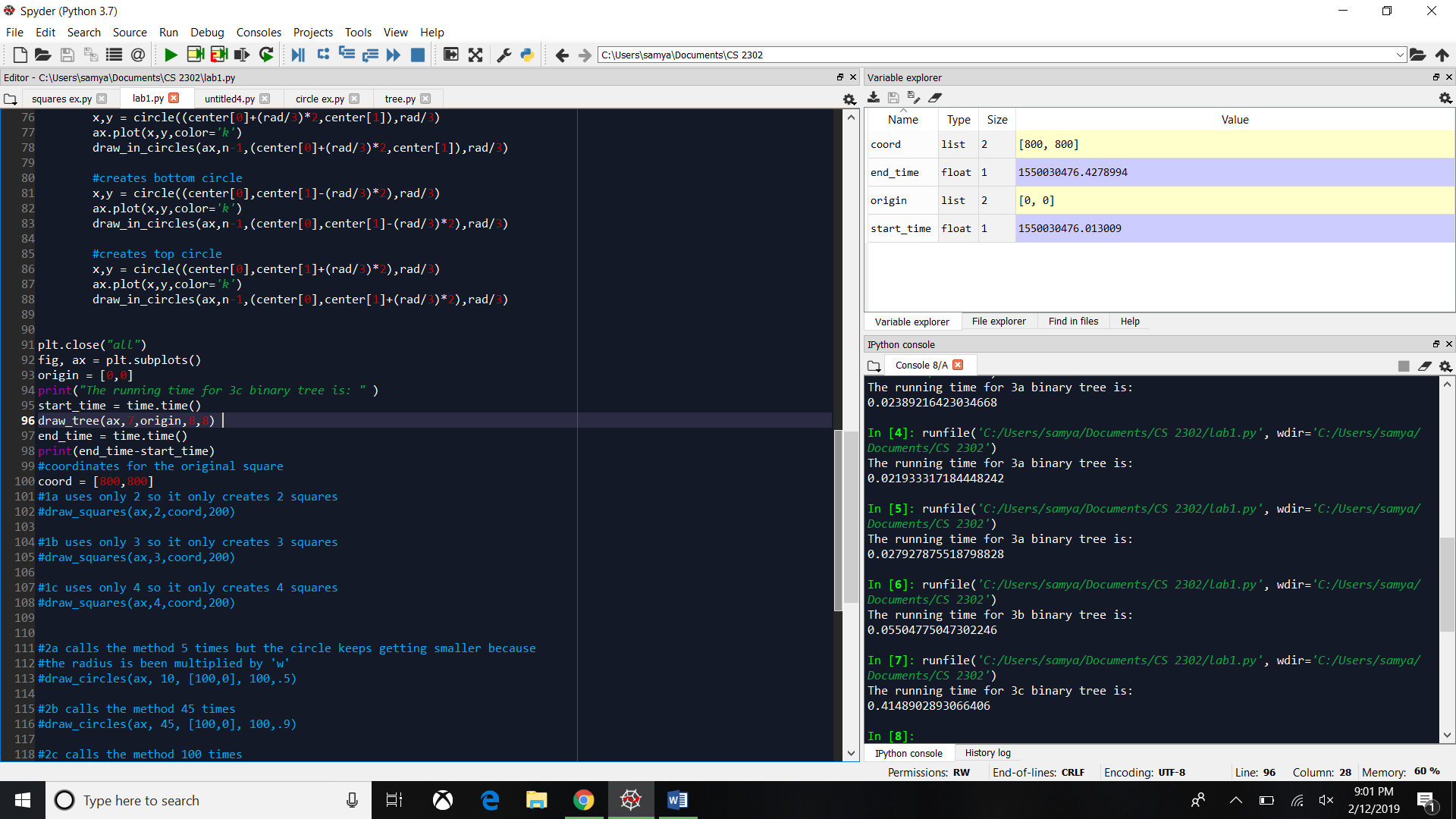


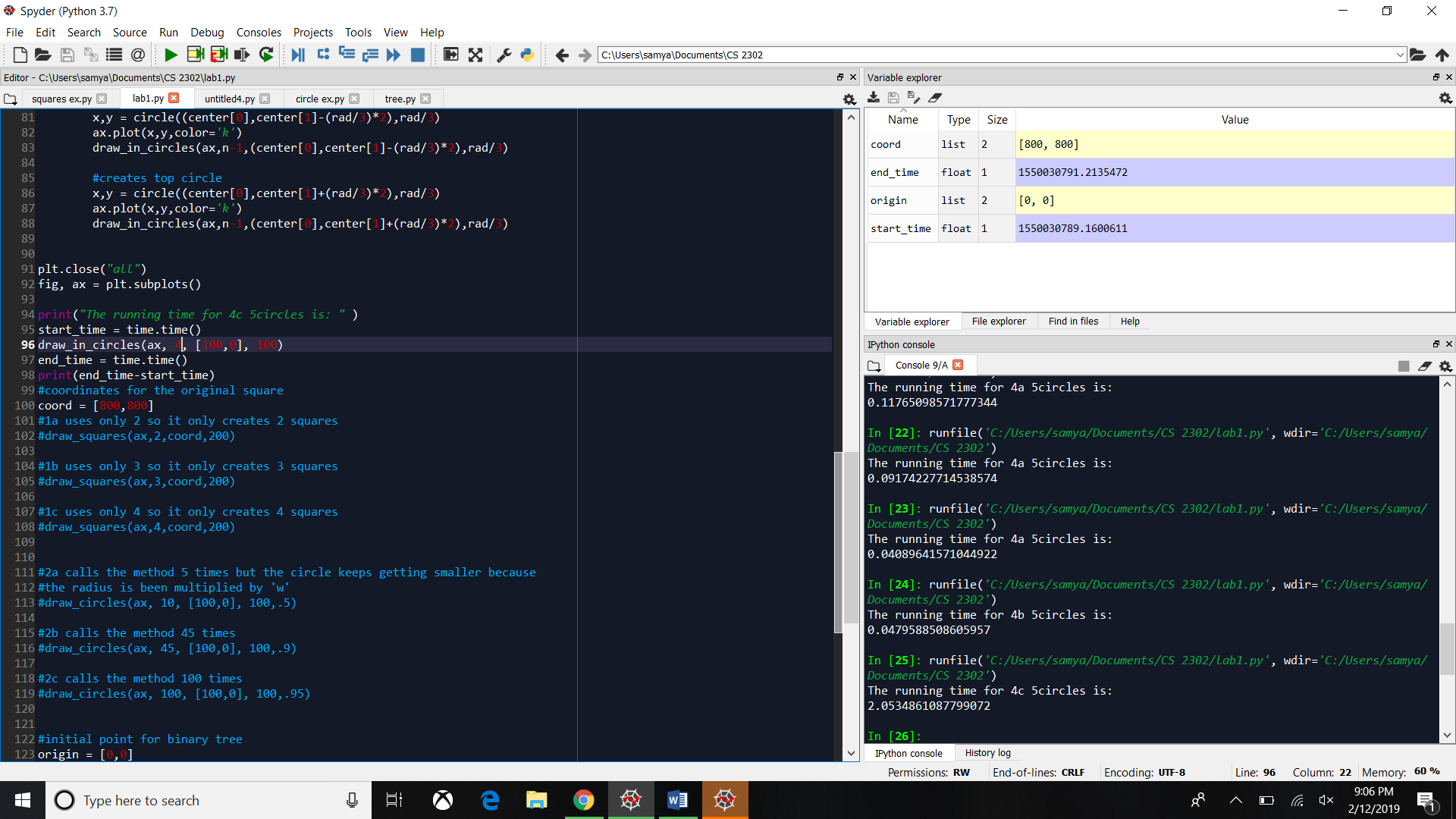
Running times for squares



Running times for circles

Running time for binary trees



Running time for 5 circles

Conclusion:

In this lab I was able to learn that it is possible to create figures using Python. Some figures are harder to make than the others and sometimes the figure does not come out correct simply because of the parameters. We always have to be careful where we add and subtract and most importantly how important the parameters are and its variables. Although it took me a long time to figure out how to create each figure I liked this kind of lab, I had never worked with creating figures and it is a new skill I learned thanks to this lab.