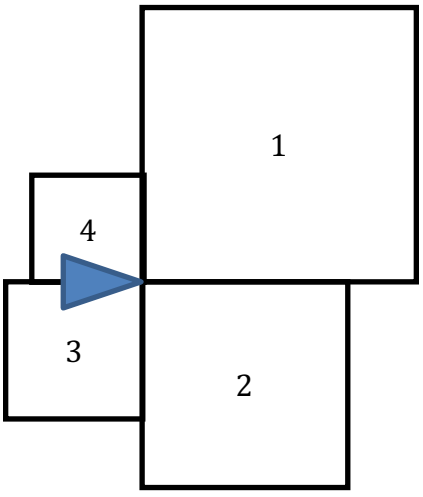
Name _____

CPADS Assignment #1

1. Develop a **strategy** with pseudocode that creates a pinwheel by obtaining user input for variables **size1, size2**, **size3**, **size4** and computing movement to the centers of each square **USING** the variables **size1**, **size2**, **size3**, **size4** in the following manners:

   a) **Square 1:** Move to the center using a forward, 90 degree left turn, forward procedure returning back to the origin after drawing the square using forward and 90 degree turn commands
   b) **Square 2:** Compute the direct distance to the center, turn 45 degrees, move forward, draw the second square, and then move back to the origin by turning around and using a forward command
   c) **Square 3:** Compute the (x,y) coordinates of the third square assuming the turtle begins at the origin and move directly to this location by assuming you have a command that can position the turtle to an absolute coordinate, and then draw the third square. **DO NOT** return to the origin.
   d) **Square 4:** Move directly from the center of the third square to the center of the fourth square**:**
      - Compute the **dx** and **dy** distances from the center of the third square to the center of the fourth square
      - Compute the distance and angle to go *directly* from one center to the other (Hint: Use the tangent function to calculate the angle)
      - Use one turn and one forward command to move to the center of the fourth square (Hint: Consider what direction the cursor is facing after finishing drawing the third square to determine the proper angle and direction to turn)
      - Draw the fourth square and assume you have a command to directly return the turtle to the origin

   Hint: ***SKETCH STRATEGIES TO FIGURE OUT THE PROPER COMPUTATIONS!***



Name _____

2. Open PyCharm making sure to select the Python 3.x interpreter. Create a new project named **CS100-Assign1**. Right click on **CS100-Assign1** in the left sidebar and select **New->Python File**. Name the file **pinwheelCompute.py**. Type the following code **exactly** as shown *copying* the **drawSquareFromCenter()** function code from **pinwheel.py** in **CS100-Lab3**

```
# import turtle graphics library
import turtle
# import math functions
from math import *

# COPY THE CODE FOR THIS FUNCTION FROM LAB 3 HERE
# Function to draw a square about the current position
#    First argument is turtle to draw with
#    Second argument is size of square sides
def drawSquareFromCenter(t,size):...

def main():
    # Create turtle for drawing named bob
    bob = turtle.Turtle()

    # Get user input for size of first square
    size1 = int(input('Enter size for first square: '))

    # Draw graphics
    # TODO: ADD CODE HERE
    drawSquareFromCenter(bob,size1)

    # Press <enter> to exit program
    input()

# Execute program
main()
```

The program should prompt the user to enter a size for the first square draw it centered about the origin.

Name _____

3. **USING YOUR STRATEGY FROM PART 1,** complete the program by adding code that obtains user input for variables **size2**, **size3**, **size4** and computes movement to the centers of each square **USING** the variables **size1**, **size2**, **size3**, **size4**. **THERE SHOULD BE NO COMPUTATIONS IN DRAWING COMMANDS, use intermediate variables for calculated values.** You **MUST** move in the following manners:

   a) **Square 1:** Move to the center using a forward, 90° left turn, forward procedure returning back to the origin after drawing the square using forward and 90° turn commands
   b) **Square 2:** Declare a variable for the direct distance to the center **dist**, turn 45°, move forward, draw the second square, and then move back to the origin by turning around and using a forward command (Hint: The square root function in python is **sqrt(x)**)
   c) **Square 3:** Declare two variables **x** and **y** and compute the (x,y) coordinates of the third square assuming the turtle begins at the origin. Use the **setposition(x,y)** turtle command to move directly to this location, and then draw the third square. **DO NOT** return to the origin.
   d) **Square 4:** Declare three variables **dx**, **dy, dist,** and **ang** computed as follows**:**
      - Compute the **dx** and **dy** distances from the center of the third square to the center of the fourth square
      - Compute **dist** and **ang** as the distance and angle to go *directly* from one center to the other (Hint: Use the tangent function to calculate the angle. In python, the tangent function is **atan(x)** which returns the angle in *radians*. To convert from radians to degrees, use the python command **degrees(x)**)
      - Use one turn and one forward command to move to the center of the fourth square (Hint: Consider what direction the cursor is facing after finishing drawing the third square to determine the proper angle and direction to turn)
      - Draw the fourth square and move back to the origin using the **home()** command

   Hint: ***EXPLAIN YOUR AGILE DEVELOPMENT APPROACH FOR YOUR PROGRAM, i.e. discuss the order in which you added code. ADD COMMENTS TO YOUR CODE EXPLAINING EACH SECTION!***