Name _____

CPADS Assignment #1 – Due Monday, 9/14 at the beginning of class

Note: There will be a quiz based on this assignment on Monday

Possible instructions

pu():	Pen Up – does not draw while moving
pd():	Pen Down – will draw when moving
fd(x):	Forward – move forward ***x*** units ('-' moves backwards)
rt(x):	Right turn – turn right ***x*** # of degrees (no movement)
lt(x):	Left turn – turn left ***x*** # of degrees (no movement)
repeat(x):	Repeat the indented instructions ***x*** # of times
incr(x,i):	Increment variable ***x*** by amount ***i*** ('-' decrements)

Examples:

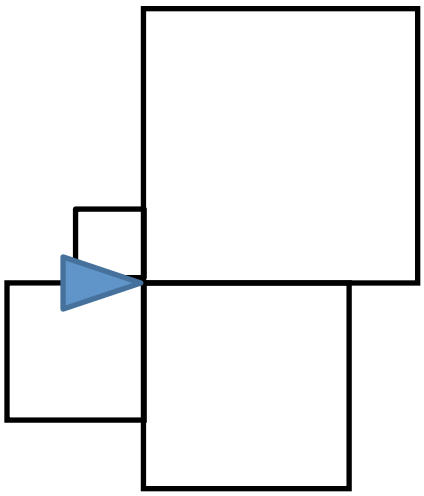
To move forward 1 unit:	**fd(1)**
To move backward 2 units:	**fd(-2)**
To turn right 60 degrees:	**rt(60)**
To repeat the indented instructions 3 times:	**repeat(3)**
To increment *x* by 2:	**incr(x,2)**
To decrement *x* by 1:	**incr(x,-1)**

1. Write a **Square(x)** function that draws a square with sides of length ***x*** in a counter-clockwise fashion, i.e. always turning left like NASCAR. Note: You may **NOT** assume that the pen is down when the function is called and you must pick the pen up at the end of the function.

Name _____

2. Using the **Square(x)** function from problem 1, create the following figure (assume the squares are sizes 1, 2, 3, 4) **WITHOUT** using any additional **repeat(x)** or **incr(x,i)** commands.

   - Check your work by showing on the figure the location and orientation of the cursor immediately before calling each **Square(x)** function.
   - Number each square in the order they are drawn.
   - If you move the cursor without drawing, i.e. after a **pu()** command, draw a curved arrow from the starting to the ending location of the movement. Be sure to indicate the final orientation of the cursor following the move.
   - After you have drawn the figure, make sure to return the cursor to its original location and in its original orientation.

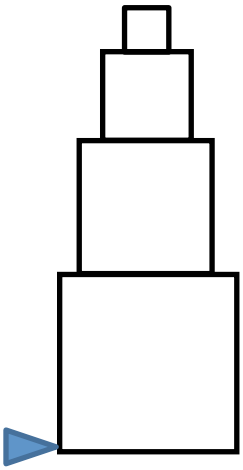


3. Rewrite your code from problem 2 to use the **repeat(x)** and **incr(x,i)** instructions to remove the repeated code.

Name _____

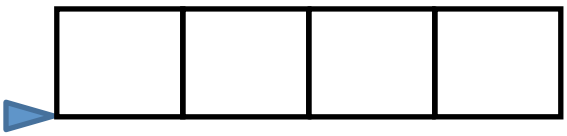
4. Write code to construct the following layer cake figure given to you to celebrate your success on problems 1 and 2! The layers are sizes 1, 2, 3, and 4 and centered on each other. You do not need to worry about where the cursor ends up once the figure is drawn.

   Hint: Consider how to draw the bottom square and position the cursor to the lower-left corner of the next higher square. Use that process along with the **repeat(x)** *and* **incr(x,i)** *instructions to add all four layers to the cake. Be sure to declare and initialize any variables you use. You can* ***decrement*** *a variable using* **incr(x,-i).**



Name _____

5. Write a function called **Row(x)** to construct a row of ***x*** boxes of size 1 using the **square(x)** (from problem 1) and **repeat(x)** commands. You **MUST** return to the cursor to the original starting position once the boxes have been drawn. Note: The example below shows what the figure would look like for **Row(4).**



Name _____

6. (Challenge) Use the **Row(x)** (from problem 5), and the **repeat(x)** and **incr(x,i)** commands to draw a pyramid with the base row having ***x*** boxes and each subsequent row having one fewer boxes until the top row only contains one box as shown in the figure below which would be the output if **x=4**. The rows are all centered on each other. Figure out how to return the cursor back to its original position relative to the number of rows that were drawn.

   As in Problem 2, check your code by executing it, and indicate the following:
   - Show on the figure the location and orientation of the cursor immediately before calling each **Square(x)** function.
   - Number each square in the order they are drawn.
   - If you move the cursor without drawing, i.e. after a **pu()** command, draw a curved arrow from the starting to the ending location of the movement. Be sure to indicate the final orientation of the cursor.