

# Problem-Based Introduction to Computer Science

2010-1

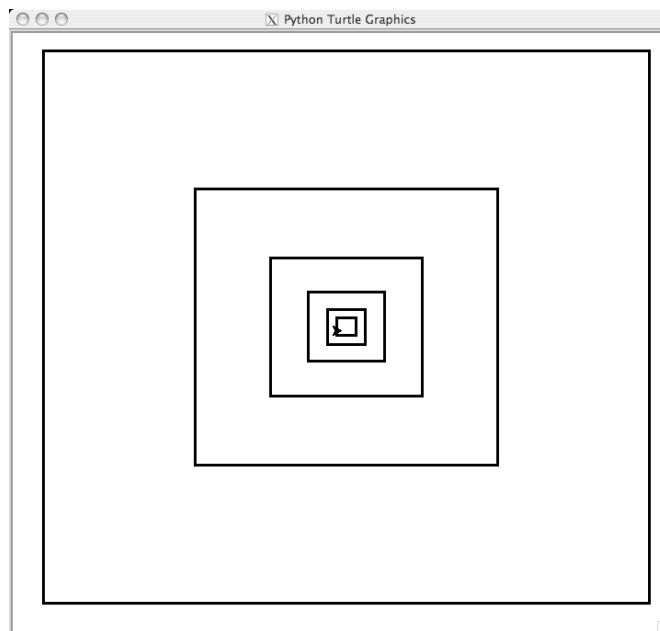
## Vanishing Square Recursion Homework 2

September 16, 2010

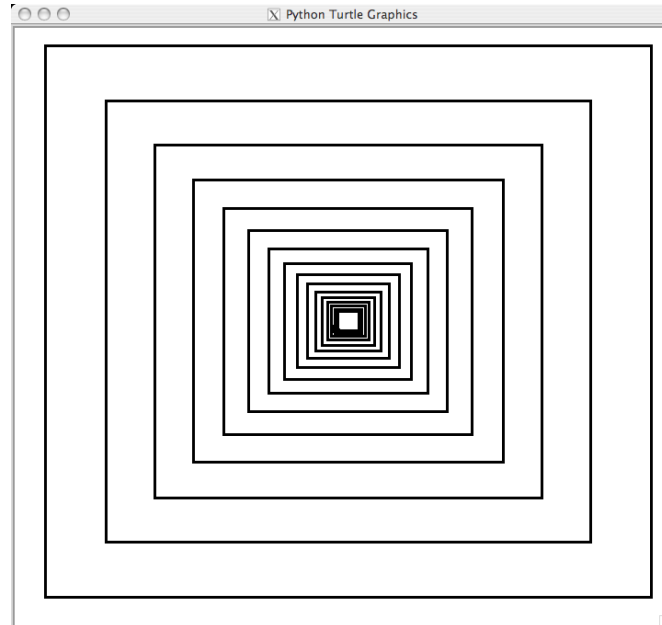
### 1 Problem Statement

The program, *vansquares.py*, creates a canvas containing a figure of nested squares drawn from the outside inward. These nested squares share a common center, the center of the drawing canvas.

The program passes the length of the outermost square to a recursive function, which draws vanishing squares and stops when the length is less than 1. In addition to the length, the program must pass a shrinkage factor: the amount by which to shrink each successive square. Below is the program output when the side length is 32 and the shrinkage scale value is 50.



With a shrinkage factor of 50, the program draws each smaller square so that it is one half the length of the next outer square; the square shrinks by 50%. A value less than 50 will produce less shrinkage (and more squares), and a larger value will produce more shrinkage (fewer squares). Below is the program output when the side length is 32 and the shrink scale value is 20.



### 1.1 Constraints on the Problem Solution

1. The solution to this problem shall use the Python *turtle* library.
2. The window canvas shall fit the drawn elements.
3. Recursion shall be the only mechanism for program sequencing.

### 1.2 Submission

You must submit your work to the appropriate homework dropbox in MyCourses. The file `vansquares.py` must contain file comments with your name, the functions that solve the problem, and a description of how to run it with different shrinkages.