Lab 1 – Review: recursion

A. Use recursion to implement a function indexOf(String text, String str) that returns the starting position of the first substring of the text that matches str. Return –1 if str is not a substring of text.

For example, indexOf("Mississippi", "sip") returns 6.

Hint: This is a bit tricky, because you must keep track of how far the match is from the beginning of the text. Make that value a parameter variable of a helper method.

Write at least 4 test cases for strings of your choice. Make sure to cover the cases where:

Substring starts at the beginning of text – returns 0
Substring is in the middle of text – returns index of stubstring
Substring is the last part of text – returns index of substring
Substring is not in text – returns -1

B. Use recursion to write a 'square root' function using the following method:

Given a value x > 0 and a guess g for the square root,
a better guess is (g + x/g) / 2.

Write a recursive helper function squareRootGuess(double x, double g).
If g^2 is approximately equal to x, return g, otherwise,
return squareRootGuess with the better guess.
(Approximately equal means: $|(x-g^2)|$ < some threshold e.g. 0.0000001)

Then write a function squareRoot(double x) that uses the helper method.
Set the initial guess to 1. (It doesn't matter; any initial value will work).

Write test cases for the square roots of {4, 9, 16, 25, and 36}
Use the 'assertAlmostEqual' method to test the result. The third argument of 'assertAlmostEqual' is the number of decimal places you want to test.

Submitting your work:
Demonstrate your work to your instructor. Make sure it's working properly and producing the expected results.