Lab04 - Recursion (and not)

Due: Fri Apr 25, 2014

Let's solve a problem recursively and then not.

- ❏ Chapter 15 Recursion

Monday - Palindromes

Complete Programming Challenge 5 on page 951… twice: recursively, and then iteratively.

In Lab04, you'll need code to:

- Read a dictionary file of words (the file will be in CSC161 format… # comment char, one word per line)
- Count the palindromes recursively
- Count the palindromes iteratively
- A `main()` to allow the user to choose a file and solution type

From the console:

- Ask the user for a dictionary file
- Ask the user if he/she wants a recursive or iterative run
- Then, run it and report the following:
    - All the palindromes found
    - The number of words in the dictionary
    - The number of palindromes found
    - The time it took to find all the palindromes

EZ, peazy.

Implementation notes

This is meatball surgery… just put everything as a static method in one class. Shoot, I'll give you this. See the k: drive. Concentrate on your palindrome algorithms.

Write down your pseudo-code… show it to me before starting on the toaster.
Also:

- Don't store the dictionary in memory. Just read it line by line. BufferedReader.readLine()
- I didn't count 1 letter words as palindromes.
- Nothing fancy on timing your solution. We'll just use the system clock. `System.currentTimeMillis()` returns the current time in milliseconds
- Recursive hint: What is the base case? What is the recursive tail?
- Iterative hint: Remember the `break` keyword? Use it to break out of a loop.
- I threw in an `enum` to show you something new/fun.