Recursion and Dynamic Programming in Sequence Analysis

Mon., Weds., Fridays @ 1pm—1:50pm, Nov. 17 to Dec 12*

While a huge variety of problem solving techniques are employed in bioinformatics, this non-credit course will explore several that are related and particularly beautiful: recursion, (self-referential processes), dynamic programming (guided deconstruction of problems into subproblems), and induction (a mathematical technique for proving the correctness of certain equations or algorithms).

After some practice computing and exploring the Fibonacci sequence and its relation to some biological systems and the recursive quicksort algorithm, we’ll cover variations of the sequence alignment problem from multiple perspectives, including both global and local alignment (as well as a variation on the idea used by BLAST). If time allows, we may explore how these ideas apply to other classic bioinformatics problems, and/or visualize biologically-inspired systems with fractals.

This class will include assignments, giving participants a chance to hone their programming skills. Additional techniques covered will include memoization and the use of data structures such as hash tables and stacks.

Prerequisites and Languages

This material requires intermediate-level programming skills, including writing and calling functions with parameters, for-loops and if-statements, and familiarity with basic data types like lists and matrices (or, lists of lists). The idea of writing a function that takes a matrix, and uses for-loops to find and return the largest element of the matrix shouldn’t sound alien to those hoping to join.

Examples and lectures will be given in the R language, though participants are free to use any language they are sufficiently comfortable with for assignments.

Information and Signup**

* Except for the Mon./Weds. before/after Thanksgiving.
** Space will be limited for this new offering.

Shawn O’Neil: shawn.oneil@cgrb.oregonstate.edu