

## Presentation

You will piece together each practical into an overall story and present this as your final project in groups. We will be doing this in the computer lab, using the projector, so you can show your programs and results directly. Simple PowerPoint/OpenOffice-Impress presentations should preferably be used where applicable. Each presentation should take about 20-30 minutes including questions.

Effectively, we would like you to present the following:

### **Course (quick) presentation (5-10min)**

- What your genome files contain
  - What species?
  - What chromosomes?
  - etc.
- What each practical's script(s) do and how they work.
  - Showing *flowcharts* / *pseudocode* / other diagrams is a good way to do this

### **Final Assignment Presentation (15-20min)**

- The results of applying your scripts to your genomes
  - The GC, nucleotide and dinucleotide frequencies for the genomes
  - The ORFs predicted using your ORF predictor. Both show some statistics for them and show some examples
    - how many?
    - How many compared to the ones you got from GLIMMER/genscan?
    - etc.
  - A distance matrix for your genomes computed using your third program from the output of your first program. That is, feed the results of the first program into the third program
  - The tree you get when computing species phylogeny from this distance matrix
  - Compare this tree with the trees you computed during the various practicals

**Final Assignment:** report analyzing the genomes based on the nucleotide frequency, GC content and distance matrices to find the evolutionary relationship between them and to find the possible ORF's from the genomes.

- DNA genome statistics tool, protein statistics tool
  - compute GC content and nucleotide (dinucleotide) frequency in a genome
  - compute amino acid (diamino acid) frequencies in a proteome

For GC content, decide whether or not to count undefined nucleotides (Ns) as part of the sequence for the purpose of computing the frequency.

For dinucleotides, both reading frames should be considered. That is, for the sequence

AGCCCAAGACACC

your results should be something like

#AG = 2/12  
#GC = 1/12  
#CC = 3/12  
#CA = 2/12  
#AA = 1/12  
#GA = 1/12  
#AC = 2/12

- Your own script - ORF finder
  - compute Open Reading Frames (ORFs) in a genome.

The input should be a genome file in FASTA format. The output should be a FASTA file with separate entries for each of the ORF gene sequences, with unique names.
- Distance matrix tool
  - compute the distance between two genomes from a DNA statistic above. Use distance matrix to create a species tree.

## Distance Methods

There are many ways of defining distances between biological objects. Most common are various sequences and the distances between them. In this assignment, you should compute the distance between two genomes as the distance between them with regards to various statistics.

A distance  $D(g1, g2)$  between two genomes  $g1$  and  $g2$  must satisfy two basic criteria:

$$D(g1, g1) = D(g2, g2) = 0$$

that is, everything will be at zero distance from itself, and

$$D(g1, g2) = D(g2, g1)$$

that is, distances are the same regardless of which direction you look at them from. These properties mean that a distance matrix will always

- have zero as the diagonal elements
- be symmetric, so that it is mirrored in the diagonal

Any function  $D(g1, g2)$  that fulfills these two conditions is a distance function.

Examples of a distance function could be the pure distance between GC-values

$$D = \sqrt{(GC \text{ of genome 1} - GC \text{ of genome 2})^2}$$

or the distance between nucleotide frequencies as

$$D = \sqrt{((G_1 - G_2)^2 + (C_1 - C_2)^2 + (A_1 - A_2)^2 + (T_1 - T_2)^2)}$$

where  $G_1$ , for instance, is the G frequency of genome 1, and the others are named correspondingly.

The expression could be extended further to use other statistics such as dinucleotide frequencies, or the angle between the frequency lists, or you could add some scaling or normalizing to make the distances better suited for the analysis.

## Final Project

**\*\*Python does square root of expression as `math.sqrt (expression)`, and square of expression as `expression**2`. To use `math.sqrt ()`, you must import `math`.**