

Algorithms in Comparative Genomics

Bielefeld University, WS 2019/20

Dr. Roland Wittler

<https://gi.cebitec.uni-bielefeld.de/teaching/2019winter/cg>

Exercise sheet 8, 04.12.2019

Exercise 1 (Common intervals)

Recall that a genome g contains a common interval c if all genes in c appear consecutively in g , and that $GS(C)$ defines the set of all genomes that contain all gene clusters (in this case common intervals) $c \in C$. Recall further the relation to the *Consecutive Ones Problem*.

Consider permutations on the set of genes $\{0, \dots, 14\}$ and the following set C of common intervals:

$$C := \{\{0, \dots, 5\}, \{3, \dots, 4\}, \{4, \dots, 8\}, \{9, \dots, 13\}, \{10, \dots, 12\}, \{11, \dots, 13\}\}$$

1. Inform yourself about PQ-trees, and use the applet provided on the lecture website to construct the PQ-tree corresponding to C .
2. How many (linear) genomes are in $GS(C)$? Can you derive a general formula?
3. Which other further common intervals that are not in C are implied by C ?
4. Which framed common intervals and nested common intervals can you find?

Exercise 2 (Minimal conflicting subsets)

Devise an algorithm in pseudo code to determine all minimal conflicting subsets of a given set of gene clusters. You can make use of the function $GS(C)$ to get the set of all genomes that contain all gene clusters $c \in C$.

Hint: Have a look at the pages 63–76 of the slides of the lecture.