## NAME

clm order – reorder clusterings conformal to inclusion structure

clmorder is not in actual fact a program. This manual page documents the behaviour and options of the clm program when invoked in mode *order*. The options **-h**, **--apropos**, **--version**, **-set**, **--nop** are accessible in all **clm** modes. They are described in the **clm** manual page.

## SYNOPSIS

**clm order [-prefix** <string> (*file multiplex prefix*)**] [-o** <fname> (*concatenated output in single file*)**]** <cluster|stack>+

## DESCRIPTION

Given a set of input clusterings **clm order** first transform it into a stack of strictly nesting clusterings. It does this by splitting clusters where necessary. It then reorders the coarsest (i.e. level-one) clustering, from large to small clusters. After that it reorders the second coarsest clustering conformally such that the first batch among its reordered clusters covers the level-one largest cluster, the second batch covers the level-one second largest cluster, and so on. Within these constraints, each batch of second-level clusters (corresponding to a single first-level cluster) is again ordered from larger to smaller clusters. This process is applied recursively throughout the entire stack of input clusters.

The input can be specified in multiple files, and a single file may contain multiple clusterings. The output is by default written as a concatenation of matrix files, the so-called stack format. Use the **-o** option to specify the output file. The stacked format can be converted to Newick format using **mcxdump(1)**. The output can be written to multiple files, one for each projected clustering, by using the **-prefix** option. By example, **-prefix P** leads to output in files named `P1`, `P2`, ..`PN`, where `N` is the number of clusters in the input, `P1` is the most fine-grained ordered clustering, and `PN` is the coarsest clustering.

## OPTIONS

**-prefix** (*<string>*)
**-o** (*<fname>*)
As decribed above.

## AUTHOR

Stijn van Dongen.

## SEE ALSO

**mclfamily(7)** for an overview of all the documentation and the utilities in the mcl family.