

**NAME**

`sccmap` – extract strongly connected components of directed graphs

**SYNOPSIS**

`sccmap` [`-dsv`] [`-outfile`] [*files*]

**DESCRIPTION**

`sccmap` decomposes digraphs into strongly connected components and an auxiliary map of the relationship between components. In this map, each component is collapsed into a node. The resulting graphs are printed to standard out. The number of nodes, edges and strongly connected components are printed to standard error. `sccmap` is a way of partitioning large graphs into more manageable pieces.

**OPTIONS**

The following options are supported:

- `-d` Preserve degenerate components of only one node.
- `-s` Do not print the resulting graphs. Only the statistics are important.
- `-S` Just print the resulting graphs. No statistics are printed.
- `-outfile` Prints output to the file *output*. If not given, `sccmap` uses stdout.
- `-v` Generate additional statistics. In particular, `sccmap` prints the number of nodes, edges, connected components, and strongly connected components, followed by the fraction of nodes in a non-trivial strongly connected components, the maximum degree of the graph, and fraction of non-tree edges in the graph.

**OPERANDS**

The following operand is supported:

- files* Names of files containing 1 or more graphs in dot format. If no *files* operand is specified, the standard input will be used.

**DIAGNOSTICS**

`sccmap` emits a warning if it encounters an undirected graph, and ignores it.

**AUTHORS**

Stephen C. North <north@research.att.com>  
Emden R. Gansner <erg@research.att.com>

**SEE ALSO**

`gc(1)`, `dot(1)`, `acyclic(1)`, `gvpr(1)`, `gvcolor(1)`, `ccomps(1)`, `tred(1)`, `libgraph(3)`