### **NAME**

bcomps – biconnected components filter for graphs

### **SYNOPSIS**

```
bcomps [ -stvx? ] [ -ooutfile ] [ files ]
```

### DESCRIPTION

**bcomps** decomposes graphs into their biconnected components, printing the components to standard output.

### **OPTIONS**

The following options are supported:

- -s No output graph is printed. Implies the -v flag.
- **-t** Print the underlying block-cutvertex tree.
- -x Each biconnected component is printed as a separate root graph.
- **-v** Prints number of blocks and cutvertices.

#### **−o** outfile

If specified, each root graph will be written to a different file with the names derived from *outfile*. In particular, if both  $-\mathbf{o}$  and  $-\mathbf{x}$  flags are used, then each block is written to a different file. If *outfile* does not have a suffix, the nth block of the ith graph is written to *outfile\_n\_i*. However, the 0th block of the 0th graph is written to *outfile*.

If *outfile* has a suffix, i.e., has the form *base.sfx*, then the files will have the same name as above, except appended with *.sfx*.

The block-cutvertex tree of ith graph is written to *outfile\_n\_T*, with an appended suffix if specified.

By default, each input graph is printed, with each block given as a subgraph whose name is a concatenation of the name of the input graph, the string "bcc" and the number of the block.

#### **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.

# **RETURN CODES**

**bcomps** returns **0** if all the input graphs are biconnected; and non-zero if any graph has multiple blocks, or any error occurred.

### **BUGS**

It is possible, though unlikely, that the names used for connected components and their subgraphs may conflict with existing subgraph names.

## **AUTHORS**

Emden R. Gansner <erg@research.att.com>

# **SEE ALSO**

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