NAME

tsort — stable total ordering

SYNOPSIS

tsort [**-w**] [*file*]

DESCRIPTION

Constructs a directed graph from pairs of whitespace-separated node relationships provided in *file* (standard input stream if "**-**", the default), then writes all unique nodes to the standard output stream, one per line, sorted according to the total ordering described by that graph, consistently with the partial ordering of the final occurrence of nodes in the input.

An input pair in the form "`a a`" indicates presence of node `a`. An input pair in the form "`a b`" indicates an `a` → `b` relationship: `a` sorts earlier than `b`; this is transitive: another "`b c`" node means that `a` → `b` → `c`, i.e. `a` sorts earlier than `b` sorts earlier than `c` (indeed, a node sorts earlier than a different node if the latter can be reached from the former).

Edges that would cause a cycle are removed, as they can interfere with preserving the input's partial ordering. A diagnostic is issued to the standard error stream with each cycle-forming node pair.

EXIT STATUS

0 If **-w** was not given.

[**0**, **124**] If **-w** was given: the status equals the number of cycles detected (truncated at the maximum).

125 if *file* couldn't be read, or contained an odd token count.

EXAMPLES

The simple case

```
$ cat nodes
A B
A F
B C
B D
D E
$ tsort nodes
A
F
B
C
D
E
```

Corresponds to this graph:

```
A → F
↓
B → C
↓
D
↓
E
```

By adding a `D` → `A` relationship, an `A` → `B` → `D` → `A` loop is created:

```
$ echo A B A F B C B D D E D A | tsort
tsort: -: breaking D -> A link
A
F
B
C
D
E
```

Corresponding to this graph:

```
A → F
↑  ↓
|  B → C
|  ↓
|←D
   ↓
   E
```

But, as the the looping `D → A` link is removed, it resolves to the same graph as the previous example. **-w** would exit *1* here.

This input:

```
$ tsort
a b c c d e
g g
f g e f
h h
^D
a
b
c
d
e
f
g
h
```

Corresponds to this graph:

```
a  c  d  h
↓     ↓
b     e
      ↓
      f
      ↓
      g
```

SEE ALSO

`lorder`(1)

STANDARDS

Conforms to IEEE Std 1003.1-2024 (“POSIX.1”).

HISTORY

Appeared in Version 7 AT&T UNIX, fully formed, as `tsort`(1) ("topological sort"), pointing at `lorder`(1), which notes:

> This brash one-liner intends to build a new library from existing ’.o’ files.
>
> ar cr library ` lorder *.o | tsort`

X/Open Portability Guide Issue 2 (“XPG2”) includes it verbatim.

Issues prior to IEEE Std 1003.1-2024 (“POSIX.1”) do not mention cycle behaviour; this issue invents **-w** and specifies cycle detection and semantics, as present-day. The **EXIT STATUS** scheme corresponds to the recommendtation in the **RATIONALE**.