Topological Sort 1

Due Date:

This assignment is due Wednesday, 11 February *at the beginning of lecture*.

Purpose

Your goal in this assignment is to write a a stand-alone Java program that mimics `/usr/bin/tsort`. That is, you should write a Java program such that

```
java TSort options
```

functions almost the same as

```
/usr/bin/tsort options
```

Precise specifications are provided below.

Specifications

Input

Your program should read from `System.in` when run as "`java TSort`" and from the file *file* when run as "`java TSort file`".

Regardless of the source, each line of input should consist of two space-separated items, each item containing no internal spaces. The entire source should consist solely of Ascii characters.

Output

Your program should write its output to `System.out`.

Unlike the standard utility /usr/bin/tsort, your program may respond to input data containing a cycle by calling `System.exit` with 2, and returning no printed output. Alternatively, your program may mimic /usr/bin/tsort exactly.

In the non-error case, your program should print one item per line in an order that is consistent with the input pairs.

Implementation

Your implementation must be entirely in that is, you must not use `Runtime.getRuntime().exec("tsort")` or the like.

Any on-line sources of algorithms consulted (such as StackExchange or online code from data structures textbooks) must be explicitly and prominently documented.

More generally, whatever the inspiration for the code you write, you must understand it entirely, and be able to explain its operation.

Efficiency

For this assignment, the asymptotic complexity of the program is not important. Write in-line javadoc comments estimating the complexity of the import methods in your code.

Hand in

⇒ Provide a complete listing of your code in "`enscript -2r`"-like format. Include in your listing compilation runs to show that your code compiles cleanly, and sample test runs.