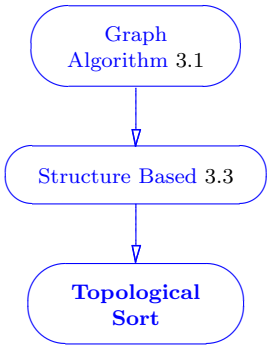
3.19 Topological-Sort

Section author: Greg Mitchell, Yogesh Girdhar, and Ramy Abdel Azim.



Refinement of: Structure Based (§3.3), therefore of Graph Algorithm (§3.1).

Prototype: `template`⟨*typename Graph, typename OutputIterator*⟩
`void topo_sort(const Graph& g, OutputIterator result)`

Effects: Creates a linear ordering of graph G such that for any edge (u, v), u precedes v in the ordering.

Asymptotic complexity: V = number of vertices, E = number of edges

- Average case (random data): $\Theta(V + E)$
- Worst case: $\Theta(V + E)$

Complexity in terms of operation counts:

V	E	Vertex Iterator Count	Adj. Edge Iterator Count
10	4	84	172
40	78	268	1060
160	1272	997	10016
640	20448	3884	132267
2560	327552	15400	2003698

Animation: `http://www.cs.odu.edu/~zeil/cgi-bin/AlgAE/cs361/toposort/toposort.cgi?htmlfile=./template.html&timeout`