



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

Interaction

[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

Tools

[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

Print/export

[Create a book](#)
[Download as PDF](#)
[Printable version](#)

Languages

[فارسی](#)
[Français](#)
[Русский](#)
[ไทย](#)
[Українська](#)
[Tiếng Việt](#)

 [Edit links](#)

[Create account](#) [Log in](#)

Article [Talk](#)

[Read](#) [Edit](#) [View history](#)

Kosaraju's algorithm

From Wikipedia, the free encyclopedia

In [computer science](#), **Kosaraju's algorithm** (also known as the **Kosaraju–Sharir algorithm**) is a [linear time algorithm](#) to find the [strongly connected components](#) of a [directed graph](#). [Aho](#), [Hopcroft](#) and [Ullman](#) credit it to an unpublished paper from 1978 by [S. Rao Kosaraju](#). The same algorithm was independently discovered by [Micha Sharir](#) and published by him in 1981. It makes use of the fact that the [transpose graph](#) (the same graph with the direction of every edge reversed) has exactly the same strongly connected components as the original graph.

Contents [\[hide\]](#)

- [1 The algorithm](#)
- [2 Complexity](#)
- [3 References](#)
- [4 External links](#)

The algorithm [\[edit\]](#)

Kosaraju's algorithm works as follows:

- Let *G* be a directed graph and *S* be an empty [stack](#).
- While *S* does not contain all vertices:
 - Choose an arbitrary vertex *v* not in *S*. Perform a [depth-first search](#) starting at *v*. Each time that depth-first search finishes expanding a vertex *u*, push *u* onto *S*.
- Reverse the directions of all arcs to obtain the transpose graph.
- While *S* is nonempty:
 - Pop the top vertex *v* from *S*. Perform a depth-first search starting at *v* in the transpose graph. The set of visited vertices will give the strongly connected component containing *v*; record this and remove all these vertices from the graph *G* and the stack *S*. Equivalently, [breadth-first search](#) (BFS) can be used instead of depth-first search.

Complexity [\[edit\]](#)

Provided the graph is described using an [adjacency list](#), Kosaraju's algorithm performs two complete traversals of the graph and so runs in $\Theta(V+E)$ (linear) time, which is [asymptotically optimal](#) because there is a matching lower bound (any algorithm must examine all vertices and edges). It is the conceptually simplest efficient algorithm, but is not as efficient in practice as [Tarjan's strongly connected components algorithm](#) and the [path-based strong component algorithm](#), which perform only one traversal of the graph.

If the graph is represented as an [adjacency matrix](#), the algorithm requires $O(V^2)$ time.

References [\[edit\]](#)

- [Alfred V. Aho](#), [John E. Hopcroft](#), [Jeffrey D. Ullman](#). *Data Structures and Algorithms*. Addison-Wesley, 1983.
- [Thomas H. Cormen](#), [Charles E. Leiserson](#), [Ronald L. Rivest](#), [Clifford Stein](#). *Introduction to Algorithms*, 3rd edition. The MIT Press, 2009. ISBN 0-262-03384-4.
- [Micha Sharir](#). A strong connectivity algorithm and its applications to data flow analysis. *Computers and Mathematics with Applications* 7(1):67–72, 1981.

External links [\[edit\]](#)

- [A description and proof of Kosaraju's Algorithm](#)↗
- [Good Math, Bad Math: Computing Strongly Connected Components](#)↗

Categories: [Graph algorithms](#) | [Graph connectivity](#)

This page was last modified on 23 September 2014, at 22:54.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

