

algorithms worth knowing

unix weapons school

A cartoon bee character with a yellow and black striped body, white wings, and a smiling face. It is positioned on the right side of the slide, appearing to fly or dance. The background is a dark blue gradient with white lightning bolts.

CT



CC3.0 share-alike attribution
copyright © 2013

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

FIXME

Recommended reading

- David Clark. “RFC 815: IP Datagram Reassembly Algorithms” (1982).
- Geoff Pike. “CityHash: Fast Hash Functions for Strings” (2012).
- Harald Prokop. “Cache-Oblivious algorithms” (1999).
- Erik Demaine. “Cache-Oblivious algorithms and data structures” (2002).
- Michael Mitzenmacher. “Cuckoo Hashing: Theory and Practice” (2007).
- Daniel Rockmore. “FFT: An algorithm the whole family can use” (2000).
- Olsson and Nilsson. “TRASH: A dynamic LC-trie and hash” (2010).
- Dan Willard. “log log range queries are possible in space $\Theta(n)$ ” (1983).
- Leonid Volnitsky. “Substring search algorithm” (2012).
- Nick Black. “LRUmap: $\mathcal{O}(1)$ massively-scalable LRU” (2010).
- Henry Warren. *Hacker's Delight (2e)* (2012).
- Donald Knuth. *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms (3e)* (1997), *Vol. 4A: Combinatorial Algorithms, Part I* (2011).

“The experiences of various groups who work on problem solving, theorem proving and pattern recognition all seem to point in the same direction: These problems are tough. There does not seem to be a royal road or a simple method which at one stroke will solve all our problems. My discussion of ultimate limitations on the speed and amount of data processing may be summarized like this: Problems involving vast numbers of possibilities will not be solved by sheer data processing quantity. We must look for quality, for refinements, for tricks, for every ingenuity that we can think of. Computers faster than those of today will be a great help. We will need them. However, when we are concerned with problems in principle, present day computers are about as fast as they ever will be.

We may expect that the technology of data processing will proceed step by step—just as ordinary technology has done. There is an unlimited challenge for ingenuity applied to specific problems.”

—Hans-Joachim Bremermann,

“Optimization through evolution and recombination” (1962).