

Main page
Contents
Featured content
Current events
Random article
Donate to Wkipedia
Wkipedia 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

Português

Pedit links

Article Talk Read Edit View histon Q

## Chaff algorithm

From Wikipedia, the free encyclopedia

**Chaff** is an algorithm for solving instances of the Boolean satisfiability problem in programming. It was designed by researchers at Princeton University, USA. The algorithm is an instance of the DPLL algorithm with a number of enhancements for efficient implementation.

## Implementations [edit]

Some available implementations of the algorithm in software are mChaff and **zChaff**, the latter one being the most widely known and used. zChaff was originally written by Dr. Lintao Zhang, now at Microsoft Research, hence the "z". It is now maintained by researchers at Princeton University and available for download as both source code and binaries on Linux. zChaff is free for non-commercial use.

## References [edit]

• M. Moskewicz, C. Madigan, Y. Zhao, L. Zhang, S. Malik. *Chaff: Engineering an Efficient SAT Solver*, 39th Design Automation Conference (DAC 2001), Las Vegas, ACM 2001.

## External links [edit]

This formal methods-related article is a stub. You can help Wikipedia by expanding it.

Categories: SAT solvers | Boolean algebra | Automated theorem proving | Constraint programming | Formal methods stubs

This page was last modified on 4 November 2014, at 20:20.

Text is available under the Oreative 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

