

NAME

parsort - Sort (big files) in parallel

SYNOPSIS

parsort *options for sort*

DESCRIPTION

parsort uses GNU **sort** to sort in parallel. It works just like **sort** but faster on inputs with more than 1 M lines, if you have a multicore machine.

Hopefully these ideas will make it into GNU Sort in the future.

EXAMPLE

Sort files:

```
parsort *.txt > sorted.txt
```

Sort stdin (standard input) numerically:

```
cat numbers | parsort -n > sorted.txt
```

PERFORMANCE

parsort is faster on files, because these can be read in parallel.

On a 48 core machine you should see a speedup of 3x over **sort**.

AUTHOR

Copyright (C) 2020 Ole Tange, <http://ole.tange.dk> and Free Software Foundation, Inc.

LICENSE

Copyright (C) 2012 Free Software Foundation, Inc.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

DEPENDENCIES

parsort uses **sort**, **bash**, **parallel**, and **mbuffer**.

SEE ALSO

sort