

## NAME

mprof – displays profiling information produced by the mpatrol library

## SYNOPSIS

mprof [options] [file]

## DESCRIPTION

The mprof command is a tool designed to read a profiling output file produced by the mpatrol library and display the profiling information that was obtained. The profiling information includes summaries of all of the memory allocations listed by size and the function that allocated them and a list of memory leaks with the call stack of the allocating function. It also includes a graph of all memory allocations listed in tabular form, and an optional graph specification file for later processing by the dot graph visualisation package.

When the PROF option is used with the mpatrol library, statistics for all dynamic memory allocations and deallocations are written to a profiling output file. The name of this file can be changed with the PROF-FILE option and the frequency at which the file is written can be specified with the AUTOSAVE option. The bounds for small allocations, medium allocations, large allocations and extra large allocations can be set with the SMALLBOUND, MEDIUMBOUND and LARGEBOUND options.

Only allocations and deallocations are recorded, with each reallocation being treated as a deallocation immediately followed by an allocation. For full memory allocation profiling support, call stack traversal must be supported in the mpatrol library and all of the program's symbols must have been successfully read by the mpatrol library before the program was run. The mprof command will attempt to compensate if either of these requirements are not met, but the displayed tables may contain less meaningful information.

The file argument must be a valid mpatrol profiling output filename but if file is omitted then mprof will use mpatrol.out as the name of the profiling output file to use. If file is given as – then the standard input file stream will be used as the profiling output file.

Much of the functionality of this tool is based upon mprof by Benjamin Zorn and Paul Hilfinger, which was written as a research project and ran on MIPS, SPARC and VAX machines. However, the profiling output files are incompatible, the tables displayed have a different format, and the way they are implemented is entirely different.

## OPTIONS

--addresses [-a]

Specifies that different call sites from within the same function are to be differentiated and that the names of all functions should be displayed with their call site offset in bytes. This affects the direct allocation and memory leak tables, as well as the allocation call graph and the graph specification file.

--call-graph [-v]

Specifies that the allocation call graph should be displayed. This is not displayed by default as it can get very large for even a moderately sized profiling output file.

--counts [-c]

Specifies that certain tables should be sorted by the number of allocations or deallocations rather than the total number of bytes allocated or deallocated. This affects the direct allocation and memory leak tables, as well as the allocation call graph and the graph specification file.

--graph-file file [-g]

Specifies that the allocation call graph should also be written to a graph specification file for later visualisation with dot. If file is given as stdout or stderr then the corresponding file stream will be used as the target for the graph specification file.

--help [-h]

Displays a quick-reference option summary.

--leaks [-l]

Specifies that memory leaks rather than memory allocations are to be written to the graph specification file. This option only affects the output from the --graph-file option.

--stack-depth depth [-n]

Specifies the maximum stack depth to use when calculating if one call site has the same call stack as another call site. This also specifies the maximum number of functions to display in a call stack. If depth is 0 then the call stack depth will be unlimited in size. The default call stack depth is 1. This affects the memory leak table.

--version [-V]

Displays the version number of the mprof command.

## SEE ALSO

mpatrol(1), mptrace(1), mleak(1), mpsym(1), mpedit(1), hexwords(1), prof(1), gprof(1), dot(1), libmpatrol(3), libmpalloc(3).

The mpatrol manual and reference card.

<http://www.cbmamiga.demon.co.uk/mpatrol/>

## AUTHOR

Graeme S. Roy <graeme.roy@analog.com>

## COPYRIGHT

Copyright (C) 1997-2001 Graeme S. Roy <graeme.roy@analog.com>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library 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 Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.