

NAME

xxhsum - print or check xxHash non-cryptographic checksums

SYNOPSIS

xxhsum [*OPTION*]... [*FILE*]...

xxhsum -b [*OPTION*]...

xxh32sum is equivalent to **xxhsum -H0**, **xxh64sum** is equivalent to **xxhsum -H1**, **xxh128sum** is equivalent to **xxhsum -H2**.

DESCRIPTION

Print or check xxHash (32, 64 or 128 bits) checksums.

When no *FILE*, read standard input, except if it's the console. When *FILE* is -, read standard input even if it's the console.

xxhsum supports a command line syntax similar but not identical to md5sum(1). Differences are:

- O **xxhsum** doesn't have text mode switch (**-t**)
- O **xxhsum** doesn't have short binary mode switch (**-b**)
- O **xxhsum** always treats files as binary file
- O **xxhsum** has a hash selection switch (**-H**)

As xxHash is a fast non-cryptographic checksum algorithm, **xxhsum** should not be used for security related purposes.

xxhsum -b invokes benchmark mode. See **OPTIONS** and **EXAMPLES** for details.

OPTIONS

-V, --version

Displays xhsum version and exits

-H*HASHTYPE*

Hash selection. *HASHTYPE* means **0**=XXH32, **1**=XXH64, **2**=XXH128, **3**=XXH3. Note that **-H3** triggers **--tag**, which can't be skipped (this is to reduce risks of confusion with **-H2 (XXH64)**).

Alternatively, *HASHTYPE* **32**=XXH32, **64**=XXH64, **128**=XXH128. Default value is **1** (XXH64)

--binary

Read in binary mode.

--tag

Output in the BSD style.

--little-endian

Set output hexadecimal checksum value as little endian convention. By default, value is displayed as big endian.

-h, --help

Displays help and exits

The following options are useful only when verifying checksums (-c):**-c, --check *FILE***

Read xxHash sums from *FILE* and check them

-q, --quiet

Don't print OK for each successfully verified file

--strict

Return an error code if any line in the file is invalid, not just if some checksums are wrong. This policy is disabled by default, though UI will prompt an informational message if any line in the file is detected invalid.

--status

Don't output anything. Status code shows success.

-w, --warn

Emit a warning message about each improperly formatted checksum line.

The following options are useful only benchmark purpose:

-b Benchmark mode. See EXAMPLES for details.

-b# Specify ID of variant to be tested. Multiple variants can be selected, separated by a ',' comma.

-BBLOCKSIZE

Only useful for benchmark mode (**-b**). See EXAMPLES for details. *BLOCKSIZE* specifies

benchmark mode's test data block size in bytes. Default value is 102400

-iITERATIONS

Only useful for benchmark mode (**-b**). See *EXAMPLES* for details. *ITERATIONS* specifies number of iterations in benchmark. Single iteration lasts approximately 1000 milliseconds. Default value is 3

EXIT STATUS

xxhsum exit **0** on success, **1** if at least one file couldn't be read or doesn't have the same checksum as the **-c** option.

EXAMPLES

Output xxHash (64bit) checksum values of specific files to standard output

```
$ xhsum -H1 foo bar baz
```

Output xxHash (32bit and 64bit) checksum values of specific files to standard output, and redirect it to **xyz.xxh32** and **qux.xxh64**

```
$ xhsum -H0 foo bar baz > xyz.xxh32
```

```
$ xhsum -H1 foo bar baz > qux.xxh64
```

Read xxHash sums from specific files and check them

```
$ xhsum -c xyz.xxh32 qux.xxh64
```

Benchmark xxHash algorithm. By default, **xxhsum** benchmarks xxHash main variants on a synthetic sample of 100 KB, and print results into standard output. The first column is the algorithm, the second

column is the source data size in bytes, the third column is the number of hashes generated per second (throughput), and finally the last column translates speed in megabytes per second.

```
$ xxhsum -b
```

In the following example, the sample to hash is set to 16384 bytes, the variants to be benched are selected by their IDs, and each benchmark test is repeated 10 times, for increased accuracy.

```
$ xxhsum -b1,2,3 -i10 -B16384
```

BUGS

Report bugs at: <https://github.com/Cyan4973/xxHash/issues/>

AUTHOR

Yann Collet

SEE ALSO

md5sum(1)