

**NAME**

`shasum` – Print or Check SHA Checksums

**SYNOPSIS**

Usage: `shasum [OPTION]... [FILE]...`

Print or check SHA checksums.

With no `FILE`, or when `FILE` is `-`, read standard input.

```
-a, --algorithm 1 (default), 224, 256, 384, 512, 512224, 512256
-b, --binary    read in binary mode
-c, --check     read SHA sums from the FILEs and check them
               --tag      create a BSD-style checksum
-t, --text     read in text mode (default)
-U, --UNIVERSAL read in Universal Newlines mode
               produces same digest on Windows/Unix/Mac
-0, --01       read in BITS mode
               ASCII '0' interpreted as 0-bit,
               ASCII '1' interpreted as 1-bit,
               all other characters ignored
```

The following five options are useful only when verifying checksums:

```
--ignore-missing don't fail or report status for missing files
-q, --quiet      don't print OK for each successfully verified file
-s, --status     don't output anything, status code shows success
               --strict  exit non-zero for improperly formatted checksum lines
-w, --warn       warn about improperly formatted checksum lines

-h, --help       display this help and exit
-v, --version    output version information and exit
```

When verifying SHA-512/224 or SHA-512/256 checksums, indicate the algorithm explicitly using the `-a` option, e.g.

```
shasum -a 512224 -c checksumfile
```

The sums are computed as described in FIPS PUB 180-4. When checking, the input should be a former output of this program. The default mode is to print a line with checksum, a character indicating type (`'*` for binary, `'` for text, `'U'` for UNIVERSAL, `'^'` for BITS), and name for each `FILE`. The line starts with a `'\'` character if the `FILE` name contains either newlines or backslashes, which are then replaced by the two-character sequences `'\n'` and `'\\'` respectively.

Report `shasum` bugs to [mshelor@cpan.org](mailto:mshelor@cpan.org)

**DESCRIPTION**

Running *shasum* is often the quickest way to compute SHA message digests. The user simply feeds data to the script through files or standard input, and then collects the results from standard output.

The following command shows how to compute digests for typical inputs such as the NIST test vector “abc”:

```
perl -e "print qq(abc) " | shasum
```

Or, if you want to use SHA-256 instead of the default SHA-1, simply say:

```
perl -e "print qq(abc) " | shasum -a 256
```

Since *shasum* mimics the behavior of the combined GNU *sha1sum*, *sha224sum*, *sha256sum*, *sha384sum*,

and *sha512sum* programs, you can install this script as a convenient drop-in replacement.

Unlike the GNU programs, *shasum* encompasses the full SHA standard by allowing partial-byte inputs. This is accomplished through the BITS option (*-0*). The following example computes the SHA-224 digest of the 7-bit message *0001100*:

```
perl -e "print qq(0001100)" | shasum -0 -a 224
```

**AUTHOR**

Copyright (c) 2003–2017 Mark Shelor <mshelor@cpan.org>.

**SEE ALSO**

*shasum* is implemented using the Perl module Digest::SHA.