

**NAME**

**wc** — line, word, character, byte, and line length count

**SYNOPSIS**

```
wc [-lwmcL] [file]...
wc [-lwmcL] --files0-from=files
```

**DESCRIPTION**

For each *file* (standard input stream if "-", the default), writes the amount of lines, words, characters, bytes, or the maximum line width, followed by the filename, if any, separated by spaces, to the standard output stream. If more than one filename was specified, a total is produced as well.

With **--files0-from**, the contents of *files* (standard input stream if "-") are used as a NUL-separated list of filenames instead.

If none of **-lwmcL** are specified, defaults to **-lwc**. The output is columnated and always in the **-lwmcL** order. **-wcL** operate according to the current locale, conversion errors are silently ignored.

If only **-c** is selected, the size is taken from `fstat(2)` for regular files, and the appropriate `ioctl(2)` for devices, if supported.

**OPTIONS**

<b>-l, --lines</b>	Number of newline ( <b>0x0A</b> ) bytes.
<b>-w, --words</b>	Number of words, defined as non-whitespace separated by whitespace in the current locale.
<b>-m, --chars</b>	Number of characters in the current locale.
<b>-c, --bytes</b>	Number of bytes read.
<b>-L, --max-line-length</b>	The widest line according to the current locale. Lines are delimited by newlines, carriage returns, and form feeds, tabs stopped at 8 columns, and non-printable characters ignored.
<b>--files0-from=files</b>	Read filenames from NUL-separated <i>files</i> instead of arguments.

**EXIT STATUS**

**1** if a file couldn't be opened or read. Diagnostics are withheld and mixed in with their corresponding output line, if any.

**EXAMPLES**

Assuming a default UTF-8 encoding:

```
$ cat form
Groceries for February:
    Bananas 3.5kg    $4.51
    Kiwis    2kg     $3.19    Call Siegfried to explain short!
    Bread                    $20.21
```

```
$ cat data
147.312$      12$      12      2.3%
11 520$ 320$   30      20%
```

```
$ wc form data
 4 16 111 form
 2  9  41 data
 6 25 152 total
```

```
$ wc -lwmcL form data
 4 16 110 111 64 form
 2  9  41  41 36 data
 6 25 151 152 64 total
```

```
$ printf '\x88' | wc -lwmcL # Invalid UTF-8: one byte, no characters
```

0 0 0 1 0

## SEE ALSO

`iswspace(3)`, `wcwidth(3)`

## STANDARDS

Conforms to IEEE Std 1003.1-2024 (“POSIX.1”). POSIX only specifies **-lwmc**, with **-m** excluding **-c**: accepting them together, **-L**, and **--files0-from** are extensions, originating from the GNU system. The column order is compatible therewith.

Be wary about assuming anything about the spacing of the numbers: some implementations columnate, usually inconsistently, others output fixed-width space-padded numbers, regardless of the fields selected. Tokenisation is required to achieve consistent results. The standard allows any blanks (but all implementations just use spaces). This implementation’s column widths match the GNU system’s — none if outputting one column for one file or if **--files0-from**, otherwise all columns are equally wide, enough to fit the sum of all byte counts (as assessed by `st_size`; non-regular bump the minimum to 7 columns).

NetBSD and FreeBSD support **-L**, but FreeBSD counts in bytes unless **-m**. Under NetBSD, **-mc** override each other. Under OpenBSD, **-m** overrides **-c**.

## HISTORY

Appears in the first edition of the UNIX Programmer’s Manual as `wc(I)`:

NAME `wc` -- get (English) word count

SYNOPSIS `wc name1 ...`

Counting words, text lines, and “**roff**” control lines — those beginning with a ‘.’. Words separated by newlines, spaces, or tabs, and counted only on text lines. The output format is much more verbose, too:

```
@ cat wc-test
.ft B
wc test
@ wc wc-test

File: wc-test
text 1 lines
control 1 lines
words 2
```

Version 3 AT&T UNIX extends control lines to start with ‘!’ and ‘”’ and outputs a total if at least two files were specified.

Version 4 AT&T UNIX reads the standard input stream if no files were specified.

Version 5 AT&T UNIX sees a new implementation in C with no total, but a familiar output format:

`lines words filename`

with words separated by anything from **0x20** (space) down and **0x7F** (DEL) up, numbers padded to seven digits, and the filename omitted if none specified (the space before it persists, however). This implementation is dated for the 27th of November, 1974 (likely an import date).

A Version 5 AT&T UNIX manual page dated 10th of March, 1974 paints a much more complex, but clearly derived from the above implementation, image:

`wc [ -rlwpc ] [ name ... ]`

With **-r** ignoring **roff** control lines, this time defined as starting with ‘.’ or ‘”’. Unless **-lwpc**, the default is **-lw**, which output the expected. **-a** counts the amount of words consisting of alphanumeric-or-underscore characters. **-p** counts the amount of words consisting of punctuation (printing-alphanumeric). It’s not clear if the underscore is alphanumeric in this case. **-c** counts the amount of **roff** control lines, regardless of **-r**.

Version 6 AT&T UNIX inherits the former implementation, with an added re-appearance of the total count.

Version 7 AT&T UNIX sports a familiar

**wc** [ **-lwc** ] [ name ... ]

**SYNOPSIS**, defaulting to **-lwc**. The fields are written in the order specified (rather, the characters supported after the **-** are, with, e.g., **-llejować** equivalent to **llw**). All three columns remain space-padded to seven characters and the space before the filename is only output if one exists. Except that **w** always has an additional space after the number.

## The BSD

4BSD sees

**wc** [ **-lwcpt** ] [ **-bbaud** ] [ **-spagesize** ] [ **-u** ] [ **-v** ] [ name ... ]

with the same parsing for **-lwcpt**. **-p** adds a field defined as **ceil(1 / pagesize)**, with a default *pagesize* of **60**. **-t** adds a field defined as **c / (baud / 10)** with an exception for *baud*=**100** yielding **10** characters per second, and default *baud* of **300**. **-u** (for UUCP) divides that by a further *10/9*ths, for "27cps at 300 baud".

Numbers are space-padded to seven characters. Time is scaled to the first of "**hr**" or "**mi**" equal to at least **1**, or "**se**" otherwise, space-padded to two digits plus one of precision, a space, the unit, and a tab. **-v** selects **lwcpt**, replaces number padding with being followed by a tab, and starts off with a tab-separated heading, with columns *lines*, *words*, *chars*, *pages*, and *time@baud*. The filename, if any, is still preceded by a space, however. An output example follows:

```
$ ./wc3 -v -u wc*
lines    words    chars    pages    time@300
11       76       18200    1        11.2 mi   wc
8        65       16936    1        10.4 mi   wc2
86       196      1356     2        50.0 se   wc2.c
15       79       17464    1        10.8 mi   wc3
195      463      5586     3        3.4 mi    wc3.c
52       101      732      1        27.0 se   wc.c
367      980      60274    9        37.2 mi   total
```

4.2BSD removes **-pt** and **-svub**, and adds a space before each number.

4.3BSD-Tahoe uses normal flag processing for **-lwc**, allows a **-** file as standard input (but also omits the filename, as in the no-filename case), and reverts back to delimiting words only by the newline, tab, and space. If only **-c** was specified, the files are *stat(2)*ted, and, if regular or directories, the size is used as the byte count.

4.4BSD notes, as it processes **-c** byte-by-byte, that:

This loses in the presence of multi-byte characters. To do it right would require a function to return a character while knowing how many bytes it consumed.

## System V

Programmer's Workbench (PWB/UNIX), otherwise inheriting from Version 6 AT&T UNIX, sees **-l**, suppressing the word count, leaving only the line count and the filename, if any, and no total.

AT&T System III UNIX, inheriting from Version 7 AT&T UNIX, fixes the extraneous space after the word count, refuses non-**lwc** columns, and exits **2** on open failure or invalid columns.

AT&T System V Release 4 UNIX becomes locale-aware and adds spaces after all numbers, preventing them melding together.

## Standards

IEEE Std 1003.2-1992 ("POSIX.2") specifies **-clw**, but is otherwise as present-day. X/Open Portability Guide Issue 4 ("XPG4") creates **-m**.