

TeXcount Quick Reference Guide

Version 2.3

1 Command line options

Syntax for running TeXcount:

```
texcount [options] [files]
```

where `texcount` refers to the TeXcount Perl-script, and the options may be amongst the following:

- v** Verbose (same as `-v3`).
- v0** No details (default).
- v1** Prints counted text, marks formulae.
- v2** Also prints ignored text.
- v3** Also includes comments and options.
- v4** Same as `-v3 -showstate`.
- showstate** Show internal states (with verbose).
- brief** Only prints a one line summary of the counts for each file.
- q, -quiet** Quiet mode, does not print error messages. Use is discouraged, but it may be useful when piping the output into another application.
- strict** Prints a warning of begin-end groups for which no specific rule is defined.
- total** Only give total sum, no per file sums.
- 1** Same as specifying `-brief` and `-total`, and ensures there will only be one line of output. If used with `-sum`, the output will only be the total number.
- 0** Same as `-1`, i.e. `-brief` and `-total`, but does not put a line shift at the end. This may be useful when the one line output is to be used by another application, e.g. Emacs, for which the line shift would otherwise need to be stripped away.
- template="..."** Specify an output template which is used to generate the summary output for each file and for the total count. Codes `{label}` is used to include values, where `label` is one of 0 to 7 (for the counts), SUM, ERROR or TITLE (first character of label is sufficient). Conditional inclusion is done using `{label?text?label}` or `{label?if non-zero|if zero?label}`. If the count contains at least two subcounts, use `{SUB|template|SUB}` with a separate template for the subcounts, or `{SUB?prefix|template|suffix?SUB}`.
- sub[=...], -subcount[=...]** Generate subcounts. Valid option values are `none`, `part`, `chapter`, `section` and `subsection` (default), indicating at which level subcounts are generated. (On by default.)
- nosub** Do not generate subcounts.

- sum[=*n*, *n*, ...]** Produces total sum, default being all words and formulae, but customizable to any weighted sum of the seven counts (list of weights for text words, header words, caption words, headers, floats, inlined formulae, displayed formulae).
- nosum** Do not generate total sum. (Default choice.)
- col** Use ANSI colour codes in verbose output. This requires ANSI colours which is used on Linux, but may not be available under Windows. On by default on non-Windows systems.
- nc, -nocol** No colours (colours require ANSI). Default under Windows.
- relaxed** Relaxes the rules for matching words and macro options.
- restricted** Restricts the rules for matching words and macro options.
- Read \LaTeX code from STDIN.
- inc** Parse included files (as separate files).
- merge** Merge included files into document (in place).
- noinc** Do not parse or merge in included files (default).
- incbib** Include bibliography in count, include bbl file if needed.
- nobib** Do not include bibliography in count (default).
- incpackage=** Include rules for a given package.
- dir[=...]** Specify working directory which will serve as root for all include files. Default (`-dir=""`) is to use the present directory. Use `-dir` to use path of the main \LaTeX document.
- enc=, -encoding=** Specify encoding to use in input (and text output).
- utf8, -unicode** Use UTF-8 (Unicode) encoding. Same as `-encoding=utf8`.
- alpha=, -alphabets=** List of Unicode character groups (or digit, alphabetic) permitted as letters. Names are separated by `,` or `+`. If list starts with `+`, the alphabets will be added to those already included. The default is `Digit+alphabetic`.
- logo=, -logograms=** List of Unicode character groups interpreted as whole word characters, e.g. Han for Chinese characters. Names are separated by `,` or `+`. If list starts with `+`, the alphabets will be added to those already included. By default, this is set to include Ideographic, Katakana, Hiragana, Thai and Lao.
- ch, -chinese, -zhongwen** Turn on Chinese mode in which Chinese characters are counted. I recommend using UTF-8, although `TEXcount` will also test other encodings (GB2312, Big5, Hz) if UTF-8 fails, and other encodings may be specified by `-encoding=`.
- jp, -japanese** Turn on Japanese mode in which Japanese characters (kanji and kana) are counted. I recommend using UTF-8, although `TEXcount` will also test other encodings (e.g. EUC-JP) if UTF-8 fails, and other encodings may be specified by `-encoding=`.
- kr, -korean** Turn on Korean mode in which Korean characters (hangul and han) are counted. I recommend using UTF-8, although `TEXcount` will also test other encodings (e.g. EUC-KR) if UTF-8 fails, and other encodings may be specified by `-encoding=`. *NB: Support for Korean is experimental.*
- kr-words, -korean-words** Korean mode in which hangul words are counted (i.e. as words separated by spaces) rather than characters. Han characters are still counted as characters. See also `-korean`. *NB: Support for Korean is experimental.*

- chinese-only, ..., -korean-words-only** As options `-chinese, ..., -korean-words`, but also excludes other alphabets (e.g. letter-based words) and logographic characters.
- char, -letter** Count letters instead of words. This count does not include spaces.
- html** Output in HTML format.
- htmlcore** Only HTML body contents.
- freq[=#]** Count individual word frequencies. Optionally, give minimal frequency required to be included in output.
- stat** Produce statistics on language usage, i.e. based on the alphabets and logograms included.
- codes** Display an overview of the colour codes. Can be used as a separate option to only display the colour codes, or together with files to parse.
- nocodes** Do not display overview of colour codes.
- opt=, -optionfile=** Reads options (command line parameters) from a specified text file. Should use one option per line. May also include TC options in the same format as specified in \LaTeX documents, but prefixed by `%` rather than `%TC:`. Blank lines and lines starting with `#` are ignored; lines starting with `\` are considered to be continuations of the previous line.
- split, -nosplit** The `-split` option, which is on by default, speeds up handling of large files by splitting the file into paragraphs. To turn it off, use the `-nosplit` option.
- showver, -nover** Include version number in output with `-showver`; use `-nover` not to show it (default).
- h, -?, -help, /?** Help.
- h=, -?=?, -help=, /?=#** Help on particular macro or group name: gives the parsing rule for that macro or group if defined. You may have to use `-incpackage=package` if the rule is defined for a specific package, and this option must be placed *before* the `-h=` option on the command line.
- ver, -version** Print version number.
- lic, -license** License information.

2 \TeX count instructions embedded in \LaTeX documents

Instructions to \TeX count can be given from within the \LaTeX document using \LaTeX comments on the format

```
%TC:instruction [name] parameters
```

where the name is use for instructions providing macro handling rules to give the name of the macro or group for which the rule applies.

ignore Indicates start of a region to be ignored. End region with the TC-instruction `endignore`.

break title Break point which initiates a new subcount. The title is used to identify the following region in the summary output.

incbib Sets bibliography inclusion, same as running \TeX count with the option `-incbib`.

2.1 Adding macro handling rules

The macro handling rules all take the format

```
%TC:instruction name parameters
```

where the name indicates the macro (with backslash) or group name for which the rule applies.

macro Defines macro handling rule for the specified macro. It takes one parameter which is either an integer or a []-enclosed array of integers (e.g. [0, 1, 0]). An integer value n indicates that the n first parameters to the macro should be ignored. An array of length n indicates that the first n parameters should be handled by the rule, and the numbers in the array specifies the parsing status (see below) with which they should be parsed. Giving the number n as parameter is equivalent to giving an array of n zeroes ([0, . . . , 0]) as zero is the parsing status for ignoring text. For all macros, also those for which no rules have been defined, options enclosed in [] between or after macros and their parameters are ignored.

macroword This defines the given macro to be counted as a certain number of words, where the number is given as the parameter.

header Define macro to be a header. This is specified as the macro rule, but has the added effect is that the header counter is increase by 1. Note, however, that you should specify a parameter array, otherwise none of the parameters will be parsed as header text. The parser status for header text is 2, so a standard header macro that uses the first parameter as header should be given the parameter [2].

breakmacro Specify that the given macro should cause a break point. Defining it as a header macro does not do this, nor is it required of a break point macro that it be a header (although I suppose in most cases of interest it will be).

group This specifies a begin-end group with the given name (no backslash). It takes two further parameters. The first parameter specifies the macro rule following `\begin{name}`. The second parameter specifies the parser status with which the contents should be parsed: e.g. 1 for text (default rule), 0 to ignore, -1 to specify a float (table, group, etc.) for which text should not be counted but captions should, 6 and 7 for inline or displayed math.

floatinclude This may be used to specify macros which should be counted when within float groups. The handling rules are specified as for `macro`, but like with `header` an array parameter should be provided and parameters that should be counted as text in floats should be specified by parsing status 3. Thus, a macro that takes one parameter which should be counted as float/caption text should take the parameter [3].

preambleinclude The preamble, i.e. text between `\documentclass` and `\begin{document}`, if the document contains one, should generally not be included in the word count. However, there may be definitions, e.g. `\title{title text}`, that should still be counted. In order to be able to include these special cases, there is a `preambleinclude` rule in which one may specify handling rules for macros within the preamble. Again, the rule is specified like the `macro` rules, but since the default is to ignore text the only relevant rules to be specified require an array.

fileinclude By default, `TEXcount` does not automatically add files included in the document using `\input` or `\include`, but inclusion may be turned on by using the option `-inc`. If other macros are used to include files, these may be specified by adding `fileinclude` rules for these macros. The specification takes one parameter: 0 if the file name should be used as provided, 1 if file type `.tex` should be added to files without a file type, and 2 if the file type `.tex` should always be added.

subst This substitutes a macro (the first parameter) with any text (the remaining option). Substitution is performed only on the present file and on the text following the instruction. Note that substitution is

performed directly on the \LaTeX code prior to parsing, and the verbose output will show the substituted text. E.g. `%TC:subst \test TEST` will cause a following `\newcommand\testTEST` to be changed into `\newcommand TESTTEST`, which \TeX count will interpret differently. Use with care!