

### File and Path Selection

<code>--binary</code>	Enable binary file searching
<code>-L</code>	Follow symbolic links ( <i>not</i> default)
<code>-g &lt;GLOB&gt;</code>	Include or exclude files and directories. Overrides any other ignore logic.
<code>--iglob &lt;GLOB&gt;</code>	Like <code>-g</code> but case-insensitive
<code>--no-ignore</code>	Don't respect any ignore files
<code>-z</code>	Search in compressed files
<code>-t &lt;T&gt;</code>	Only search for <code>(-t)</code> /
<code>-T &lt;T&gt;</code>	exclude <code>(-T)</code> files of TYPE. Use <code>--type-list</code> to show all

Globbing rules match `.gitignore` style. Precede glob with a `!` to exclude.

There are several other options to not respect only certain ignore files:

`--no-ignore-dot`,  
`--no-ignore-global`,  
`--no-ignore-parent`, and  
`--no-ignore-vcs`.

The `-z` option requires decompression binaries to be on the PATH. Consider instead the `ripgrep-all` variant.

### Matcher Selection

<code>-F</code>	Treat pattern as literal string
<code>-P</code>	Use the PCRE2 regex engine
<code>--auto-hybrid-regex</code>	Choose regex engine based on features used in pattern

Use the PCRE2 engine to support look-around or backreferences.

### Matching Control

<code>-s</code>	Search case-sensitively
<code>-S</code>	Search case-insensitively if pattern is all lowercase, otherwise sensitively
<code>--crlf</code>	Treat CRLF as line terminator for <code>\$</code>
<code>--pre &lt;CMD&gt;</code>	Preprocess file with CMD before searching
<code>--pre-glob &lt;GLOB&gt;</code>	Used to specify which files to preprocess
<code>-e</code>	Used to specify multiple patterns or protect patterns starting with <code>"-</code>
<code>-w</code> , <code>--word-regex</code>	Only matches surrounded by word boundaries
<code>-c</code>	Count matched lines/file
<code>--count-matches</code>	Count matches per file

### Other Options

<code>--no-messages</code>	Hide file open/read errs
<code>--block-buffered</code>	Flush on in-memory buffer size (default in pipeline or to file)
<code>--line-buffered</code>	Flush per line (default in terminal)
<code>--encoding &lt;ENC&gt;</code>	Encoding for reading files (e.g. <code>utf-8</code> , <code>utf-16</code> , <code>ascii</code> , <code>windows-1252</code> )
<code>-j &lt;NUM&gt;</code>	Approx threads to use
<code>-[A B C] &lt;num&gt;</code>	Show <code>&lt;num&gt;</code> lines after each match (A), before (B), or before and after (C)

More encodings [here](#).

### Output Options

<code>--color &lt;WHEN&gt;</code>	<code>auto</code> , <code>never</code> , <code>always</code> , or <code>ansi</code> (emit ANSI escapes).
<code>--column</code>	Show column numbers (1-based) for first match
<code>-l</code>	Show only names of matching files
<code>--files-without-match</code>	Show only names of <i>non</i> -matching files
<code>--no-heading</code>	File path at start of line instead of before group
<code>-I</code>	Never print file name
<code>-N</code>	Suppress line numbers
<code>-0</code> , <code>--null</code>	Follow any file path with a NUL byte
<code>--null-data</code>	Use NUL as a line terminator instead of <code>"\n"</code>
<code>-o</code>	Print only match
<code>--path-separator &lt;S&gt;</code>	Alt char for outputting path separator
<code>-p</code> , <code>--pretty</code>	Alias for <code>--color</code> <code>always</code> <code>--heading</code> <code>--line-number</code>
<code>-r &lt;TEXT&gt;</code>	Replace match with <code>&lt;TEXT&gt;</code> in output (will <i>never</i> modify files)
<code>--sort &lt;SORTBY&gt;</code>	Sorts results ascending ( <code>--sortr</code> for descending). Options: <code>path</code> , <code>modified</code> , <code>accessed</code> , <code>created</code> , <code>none</code>
<code>--trim</code>	Trim lead whitespace
<code>-q</code>	Quiet, no output



By njones  
[cheatography.com/njones/](https://cheatography.com/njones/)

Published 21st February, 2020.  
 Last updated 2nd May, 2020.  
 Page 1 of 1.

Sponsored by **Readable.com**  
 Measure your website readability!  
<https://readable.com>