

**NAME**

regexp-assemble – Assemble a list of regular expressions from a file

**SYNOPSIS**

```
regexp-assemble -abcdfinprsStTuUvw file [...]
```

**DESCRIPTION**

Assemble a list of regular expression either from standard input or a file, using the `Regexp::Assemble` module.

**OPTIONS**

- a** look Ahead. Insert `(?= . . )` zero-width lookahead assertions in the pattern, where necessary.
- b** Blank. Ignore blank lines.
- c** Comment. Basic comment filtering. Strip off perl/shell comments `(\s*#. *$/)`.
- d** Debug. Turns on debugging output. See `Regexp::Assemble` for suitable values.
- i** Indent. Print the regular expression using and indent of `n` to display nesting. A.k.a pretty-printing. Implies **-p**.
- n** No newline. Do not print a newline after the pattern. Useful when interpolating the output into a templating system or similar.
- p** Print. Print the pattern. This is the default, however, it is required when the **-t** switch is enabled (because if you want to test patterns ordinarily you don't care what the assembled pattern looks like).
- r** Reduce. The default behaviour is to reduce the assembled pattern. Enabling this switch causes the reduction algorithm to be switched off. This can help you determine how much reduction is performed.

```
regexp-assemble pattern.file | wc
# versus
regexp-assemble -r pattern.file | wc
```

- s** Statistics. Print some statistics about the assembled pattern. The output is sent to `STDERR` (in order to allow the generated pattern to be redirected elsewhere).
- S** Statistics only. Like **-s**, except that the pattern itself is not output. Useful with **-d 8** to see the time taken.
- t** Test. Test the assembled expression against the contents of a file. Each line is read from the file and is matched against the pattern. Lines that fail to match are printed. In other words, no output is good output. In this mode of operation, error status is 1 in the case of a failure, 0 if all lines matched.
- T** Time. Print statistics on the time taken to reduce and assemble the pattern. (This is merely a lazy person's synonym for **-d 8**).
- u** Unique. Carp if duplicate patterns are found.
- U** Unroll. Transform `a+ et al` into `aa*` (which may allow additional reductions).
- v** Version. Print the version of the `regexp-assemble` script.
- w** Word/Whole. When testing the contents of a file with **-t**, bracket the expression with `^` and `$` in order to match the whole word or line from the file.

**DIAGNOSTICS**

Will print out a summary of the problem if an added pattern causes the assembly to fail.

**SEE ALSO**

`Regexp::Assemble`

**AUTHOR**

Copyright (C) 2004–2008 David Landgren. All rights reserved.

**LICENSE**

This script is free software; you can redistribute it and/or modify it under the same terms as Perl itself.