

This file documents GNU Spell, a clone of Unix 'spell'.

Copyright (C) 1996 Free Software Foundation, Inc.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Foundation.

File: spell.info, Node: Top, Up: (dir)

This file documents GNU Spell, a clone of Unix 'spell'. This is edition 1.0 of the GNU Spell manual, last updated on April 6, 1996.

* Menu:

- * Overview:: Preliminary information.
- * Invoking Spell:: How to run Spell.
- * Example:: An example run.
- * Problems:: Reporting bugs.
- * Concept Index:: Index of concepts.

File: spell.info, Node: Overview, Next: Invoking Spell, Up: Top

Overview

GNU Spell is a spell checking program which prints each misspelled word on a line of its own. It is designed as a clone of the standard Unix 'spell' program, and implemented as a wrapper for Ispell. Spell accepts as its arguments a list of files to read from. Within that list, the magical file name '-' causes Spell to read from standard input. In addition, when called with no file name arguments, Spell assumes that it should process standard input.

Spell is most useful when used in conjunction with other programs, because its output is minimalistic. Human users are usually more comfortable with verbose output. *note The Ispell Manual: (ispell)Top, for an screen-oriented spelling checker.

File: spell.info, Node: Invoking Spell, Next: Example, Prev: Overview, Up: Top

Invoking Spell

The format for running GNU Spell is:

```
spell [OPTION]... [FILE]...
```

Spell supports the following options:

'--version'

'-V'

Print the version number of Spell on the standard error output and then exit.

'--british'

'-b'

Use the British dictionary rather than American. Unavailable unless this dictionary was installed with Ispell.

'--dictionary=FILE'

'-d FILE'

Use the named dictionary.

'--help'

'-h'

Print an informative help message describing the options and then exit.

'--ispell=PROGRAM'

'-i PROGRAM'

Call PROGRAM as Ispell.

'--all-chains'

'-l'

Ignored; for compatibility.

'--number'

'-n'

Print the line number of each misspelled word along with the word itself.

'--print-file-name'

'-o'

Print the file name which contained the misspelled words on each line before the word.

'--stop-list=FILE'

'-s FILE'

Ignored; for compatibility.

'--verbose'

'-v'

When a word is not found in its literal form in the dictionary, it is printed.

'--print-stems'

'-x'

Ignored; for compatibility.

File: spell.info, Node: Example, Next: Problems, Prev: Invoking Spell, Up: Top

Example

Using GNU Spell is quite simple. Here's an example, which you can try when in the directory of Spell's source code.

```
$ spell --print-file-name --number sample
sample:1: Tihs
sample:1: si
sample:1: fo
sample:1: Splel
sample:1: worsd
sample:1: spellled
sample:1: worng
sample:2: fi
sample:2: cna
sample:2: dael
sample:2: fiel
sample:2: Foo
sample:2: bza
sample:2: baz
```

The words printed on standard output are all the (purposely) misspelled words in 'sample'. Notice the options '--print-file-name' and '--number' (*note Invoking Spell::.). They are responsible for the prefix you see on each line, 'sample:LINENO '.

File: spell.info, Node: Problems, Next: Concept Index, Prev: Example, Up: Top

Problems

If you find a bug in GNU Spell, or have any questions, comments, or suggestions about it, please send electronic mail to 'bug-gnu-utils@prep.ai.mit.edu'. When sending bug reports, include the version number, which you can find by running 'spell --version'. Also include in your message the operating system you are using, the output that the program produced, the output you expected, which options you invoked Spell with, and any other details that you think may be useful. Keep in mind that facts are more usually more helpful than guesses.