NAME

ndiff − compare putatively similar files, ignoring small numeric differences

SYNOPSIS

ndiff [**−?**] [**−abserr** *abserr*] [**−author**] [**−copyright**] [**−fields** *n1a-n1b,n2,n3a-n3b,...*] [**−help**] [**−logfile** *filename*] [**−minwidth** *nnn*] [**−outfile** *filename*] [**−precision** *number-of-bits*] [**−quick**] [**−quiet**] [**−relerr** *relerr*] [**−separators** *regexp*] [**−silent**] [**−version**] [**−www**] *infile1 infile2*

DESCRIPTION

When a numerical program is run in multiple environments (operating systems, architectures, or compilers), assessing its consistency can be a difficult task for a human, since small differences in numerical output values are expected.

Application of a file differencing utility, such as POSIX/UNIX **diff**(1), will generally produce voluminous output, often longer than the original files, and is thus not useful. The lesser-known UNIX **spiff**(1) utility, while capable of handling numeric fields, suffers from excessively-long running times, and often terminates prematurely.

ndiff provides a solution to this problem. It compares two files that are expected to be identical, or at least, numerically similar. It assumes that lines consist of whitespace-separated fields of numeric and non-numeric data.

A hyphen (minus sign) can be used in place of either input filename to represent *stdin*, allowing one input stream to come from a UNIX pipe. This is a common, but by no means universal, idiom in UNIX software as a workaround for the regrettable lack of standard names for the default *stdin* and *stdout* streams. On some, but not all, UNIX systems, *stdin* can be named explicitly as */dev/stdin* or */dev/fd/0*.

The default field separator characters can be modified with the **−separators** *regexp* command-line option, so that **ndiff** can also handle files with, e.g., parenthesized complex numbers, and comma-separated numbers from Fortran list-directed output. However, because line breaking and use of repeats counts in Fortran list-directed is implementation dependent, such files are not really suitable for cross-implementation file comparisons, unless the lists are kept short enough to fit on a single line.

ndiff expects the files to contain the same number of lines; otherwise, a diagnostic will be issued. Unlike **diff**(1), this program cannot handle inserted or deleted lines.

Also unlike **diff**(1) (unless **diff**'s **−b** and **−w** options are used), whitespace is *not* significant for **ndiff**, except that it normally separates fields.

Lines that differ in at least one field (as determined by the absolute and/or relative tolerances, for numeric values, or string comparisons otherwise) are reported on *stdout* in a **diff**(1) -style listing of the form

> *nnn*c*nnn*
> < line from *infile1*
> --- field *n* absolute error *x.xxe-xx* relative error *x.xxe-xx* [*nn*∗(machine epsilon)]
> > line from *infile2*

The first of these lines shows the line number twice, separated by the letter *c* (for change). The second and fourth lines begin with a two-character identifying prefix. The third, separator, line shows the field number at which the difference was found; fields beyond that one may also differ, but have not been checked. If the differing field is numeric, then the errors found are also shown on that line. If the relative error is not too big, its value is also shown as a multiple of the machine epsilon.

ndiff recognizes the following patterns as valid numbers. In the patterns, *#* is a string of one or more decimal digits, optionally separated by a nonsignificant underscore (as in the Ada programming language), *s* is an optional + or - sign, and *X* is an exponent letter, one of *D*, *d, E*, *e, Q*, or *q*:

```
s#      s#s#    s#Xs#        s#.     s#.s#   s#.Xs#
s#.#    s#.#s#  s#.#Xs#      s.#     s.#s#   s.#Xs#
```

The rigorous programming rule that determines whether a string is interpreted as a floating-point value is that it must match this very complicated regular expression (the line breaks are for readability only):

```
"^[-+]?([0-9](_?[0-9])*([.]?([0-9](_?[0-9])*)*)?|
    [.][0-9](_?[0-9])*+)
    ([DdEeQq]?[-+]?[0-9](_?[0-9])*)?$"
```

Thus, *123*, *-1q-27*, *.987d77*, *3.14159_26535_89793_23846*, and *.456-123* are all valid numbers.

Notably absent from this list are Fortran-style numbers with embedded blanks (blanks are not significant in Fortran, except in string constants). If your files contain such data, then you must convert them to standard form first, if you want **ndiff** to perform reliably. In the interests of interlanguage data exchange, most modern Fortran implementations do not output floating-point numbers with embedded spaces, so you should rarely need such file conversions.

From version 2.00, **ndiff** also recognizes patterns for optionally-signed NaN (Not-a-Number):

```
NaN     SNaN    QNaN    NaNS    NaNQ    ?.0e+0  ??.0

+NaN    +SNaN   +QNaN   +NaNS   +NaNQ   +?.0e+0 +??.0

-NaN    -SNaN   -QNaN   -NaNS   -NaNQ   -?.0e+0 -??.0
```

and optionally-signed Infinity:

```
Inf     Infinity        +.+0e+0 +.+0

+Inf    +Infinity       +.+0e+0 +.+0

-Inf    -Infinity       -.-0e+0 -.-0
```

Lettercase is *not* significant in these values.

The rigorous programming rule for whether a field is a NaN or an Infinity is determined by these complex regular expressions (again, the line breaks are for readability only):

```
"^[-+]?([QqSs]?[Nn][Aa][Nn][QqSs]?|
    [?]+[.][?0]+[DdEeQq][-+]?[0-9]+|
    [?]+[.][?0]+)$"
"^(-[Ii][Nn][Ff]|
    -[Ii][Nn][Ff][Ii][Nn][Ii][Tt][Yy]|
    -+[.][-]0+[DdEeQq][-+]?[0-9]+|
    -+[.][-]0+)$"
"^([+]?[Ii][Nn][Ff]|
    [+]?[Ii][Nn][Ff][Ii][Nn][Ii][Tt][Yy]|
    [+]+[.][-]0+[DdEeQq][-+]?[0-9]+|
    [+]+[.][-]0+)$"
```

Even though in numerical computations, a NaN is never equal to anything, even itself, for **ndiff**, fields that match a NaN pattern *are* considered equal.

Fields that match Infinity patterns are considered equal if they have the same sign.

ndiff terminates with a success exit code (on UNIX, 0) if no differences (subject to the absolute and/or relative tolerances) are found. Otherwise, it terminates with a failure exit code (on UNIX, 1).

OPTIONS

Command-line options may be abbreviated to a unique leading prefix, and letter case is ignored.

To avoid confusion with options, if a filename begins with a hyphen, it must be disguised by a leading absolute or relative directory path, e.g., */tmp/-foo.dat* or *./-foo.dat*.

GNU- and POSIX-style options of the form **--name** are also recognized: they begin with two option prefix characters.

-? Display brief usage information on *stderr* and exit with a success status code before processing any input files.

This is a synonym for **-help**.

–abserr *abserr*
Specify a maximum absolute difference permitted before fields are regarded as different. Unless the fields are all of the same approximate magnitude, you probably do not want to use this option.

A zero value for this option suppresses reports of absolute error differences.

This option may be abbreviated **–a**.

For readability, this option may also be called **–absolute-error**, or any unique prefix thereof.

–author
Show author information on *stderr* and exit with a success status code before processing any input files.

–copyright
Show copyright information on *stderr* and exit with a success status code before processing any input files.

–fields *n1a-n1b,n2,n3a-n3b,...*
By default, all fields are compared, but this option can specify a comma-separated list of numbers, and/or ranges, selecting the fields that are to be compared.

Fields are numbered starting from 1.

A field range is a pair of numbers, separated by one or more hyphens (minus signs): 4-7 and 4--7 are equivalent to 4,5,6,7.

To prevent long range-expansion loops, field ranges are restricted to a non-negative span of no more than 100: 8-8 and 1-100 are acceptable, but 3-, -5, 8-7 and 1-101 all generate an error.

–help
Display brief usage information on *stderr* and exit with a success status code before processing any input files.

This is a synonym for **–?**.

–logfile *filename*
Redirect warning and error messages from *stderr* to the indicated filename. This option is provided for user convenience on poorly-designed operating systems (e.g., IBM PC DOS) that fail to provide for redirection of *stderr* to a specified file.

This option can also be used for discarding messages, with, e.g., on UNIX systems, **–logfile** */dev/null*.

–minwidth *nnn*
Specify a minimum field width required for numeric fields containing a decimal point and/or exponent. If both such fields being compared are shorter than this, they are treated as equal.

This option is useful when fields contain relative error values given to only a few digits; such values might differ widely between two files, but those differences can be made irrelevant by invoking this option.

For readability, this option may also be called **–minimum-width**, or any unique prefix thereof.

–outfile *filename*
Redirect output from *stdout* to the indicated filename. This option is provided for user convenience on operating systems that fail to provide for redirection of *stdout* to a specified file.

–precision *number-of-bits*
Specify the number of bits in the significands used in multiple-precision arithmetic. The corresponding number of decimal digits is *floor(number-of-bits / lg 10) = floor(number-of-bits / 3.32)*.

You can use the **–version** option to see the value of the corresponding machine epsilon (the smallest number, which, when added to one, still differs from one).

The multiple-precision arithmetic library used by **ndiff** increases its working precision in multiples of a certain implementation-dependent size, usually 64 bits, so the reported machine epsilon may not decrease until *number-of-bits* has been increased beyond the next multiple of that size.

If **ndiff** was compiled without support for multiple-precision arithmetic, use of this option will elicit a warning.

−quick Suppress reading of the initialization files, *$LIBDIR/.ndiffrc*, *$HOME/.ndiffrc*, and *./.ndiffrc*. *LIBDIR* represents the name of the **ndiff** installation directory; it is not a user-definable environment variable.

Normally, the contents of those files, if they exist, are implicitly inserted at the beginning of the command line, with comments removed and newlines replaced by spaces. Thus, those files can contain any **ndiff** options defined in this documentation, either one option, or option/value pair, per line, or with multiple options per line. Empty lines, and lines that begin with optional whitespace followed by a sharp (#) are comment lines that are discarded.

If the initialization file contains backslashes, they must be doubled because the text is interpreted by the shell before **ndiff** sees it.

−quiet The maximum absolute and relative errors, and their locations, in *matching* lines are tracked, and at termination, a two-line report with their values is normally printed on *stdout*. This option suppresses that report.

This option may be abbreviated **−qui**, **−qu**, or **−q**.

−relerr *relerr* Specify a maximum relative difference permitted before fields are regarded as different. The relative error of two fields *x* and *y* is defined to be:

0	if *x* is identical to *y*, or else
abs(*x*-*y*)/min(abs(*x*),abs(*y*))	if *x* and *y* are nonzero, or else
1	if *x* is zero, and *y* is nonzero, or else
1	if *y* is zero, and *x* is nonzero, or else
0	since both *x* and *y* are zero.

This complex definition of relative error ensures that the results will be independent of the order of the two input files on the command line.

A zero value for this option suppresses reports of relative error differences.

For readability, this option may also be called **−relative-error**, or any unique prefix thereof.

If neither **−abserr** nor **−relerr** is specified, then **−relerr** *x* is assumed, where *x* is the larger of 1.0e-15 and eight times the machine epsilon (the smallest number whose sum with 1.0 still differs from 1.0).

If the specified relative error value is greater than or equal to 1.0, it is multiplied by the machine epsilon. Thus, you can specify **−relerr** *16* to allow relative errors of up to 4 bits (since 2^4 == 16).

ndiff will issue a warning if you specify a relative error value smaller than the machine epsilon, but will accept and use your specified value.

−separators *regexp* The argument is an **awk**(1) regular expression that specifies an alternate set of characters separating fields in input lines.

By default, this is a single blank, which has a special meaning in **awk**(1): leading and trailing whitespace (blanks and tabs) is first stripped, then runs of consecutive whitespace are collapsed to a single space, and finally, the line is split into fields at the spaces.

If the input files contain parenthesized complex numbers, or comma-separated numbers from Fortran list-directed output, then you should specify **−separators** *'[\t,()]'* so that blanks, tabs, commas, and parentheses separate input fields.

−silent Suppress the output of the difference lines on *stdout*.

Using both **−quiet** and **−silent** guarantees that nothing is printed on *stdout*, but the **ndiff** exit code can still be used for testing for a successful comparison.

This option may be abbreviated **−s**.

−version Show version and precision information on *stderr* and exit with a success status code before processing any input files.

The machine epsilon reported in this output may depend on a preceding **−precision** *number-of-bits* specification.

−www Show the World-Wide Web master archive location for this program on *stderr* and exit with a success status code before processing any input files.

CAVEATS

This implementation of **ndiff** can be built with support for double-precision, quadruple-precision, or multiple-precision arithmetic. The **−version** option reports the particular choice at your site. Thus, **ndiff** will not correctly handle absolute and relative error tolerances that are smaller than those corresponding to the machine epsilon in the arithmetic for which it was built, and for that reason, installers are encouraged to build the multiple-precision version, so that users can select any required precision.

WISH LIST

It would be nice to have **ndiff**'s abilities incorporated into the GNU **diff**(1) program; that way, numeric fields could be successfully compared even in files with inserted or deleted lines, and much of the entire computing world could benefit.

Perhaps some community-minded and clever reader of this documentation will take up this challenge, and present the Free Software Foundation with an improved **diff**(1) implementation that offers support for tolerant differencing of numeric files, using **ndiff** as a design model, sample implementation, and testbed!

Ideally, such an improved **diff**(1) implementation should handle numbers of *arbitrary* precision, allowing comparisons of numeric output from systems that support high-precision arithmetic, such as Lisp and symbolic algebra languages. In addition, it might choose to do its arithmetic in decimal floating-point, so as to avoid inaccuracies introduced by vendor-dependent libraries for decimal-to-native-base number conversion.

The **awk**(1) prototype version of **ndiff** supports only double-precision arithmetic; the C version is more flexible.

FILES

In the following, *LIBDIR* represents the name of the **ndiff** installation directory; it is not a user-definable environment variable. If **ndiff** has been installed properly at your site, the value of *LIBDIR* is

```
/usr/local/share/lib/ndiff/ndiff-2.00
```

`$LIBDIR/.ndiffrc` System-specific initialization file containing customized **ndiff** command-line options.

`$HOME/.ndiffrc` User-specific initialization file containing customized **ndiff** command-line options.

`./.ndiffrc` Current-directory-specific initialization file containing customized **ndiff** command-line options.

`$LIBDIR/ndiff.awk` **awk**(1) program invoked by **ndiff**. This file will not be installed if the C version of **ndiff** was built.

SEE ALSO

awk(1), **bawk**(1), **cmp**(1), **diff**(1), **gawk**(1), **mawk**(1), **nawk**(1), **spiff**(1).

AUTHOR

Nelson H. F. Beebe
Center for Scientific Computing
University of Utah
Department of Mathematics, 322 INSCC
155 S 1400 E RM 233
Salt Lake City, UT 84112-0090
USA
Email: `beebe@math.utah.edu`, `beebe@acm.org`,
`beebe@computer.org`, `beebe@ieee.org` (Internet)
WWW URL: `http://www.math.utah.edu/~beebe`
Telephone: +1 801 581 5254
FAX: +1 801 585 1640, +1 801 581 4148

AVAILABILITY

ndiff is freely available; its master distribution can be found at

```
ftp://ftp.math.utah.edu/pub/misc/
http://www.math.utah.edu/pub/misc/
```

in the file *ndiff-x.yy.tar.gz* where *x.yy* is the current version. Other distribution formats are usually available at the same location.

That site is mirrored to several other Internet archives, so you may also be able to find it elsewhere on the Internet; try searching for the string *ndiff* at one or more of the popular Web search sites, such as

```
http://altavista.digital.com/
http://search.microsoft.com/us/default.asp
http://www.dejanews.com/
http://www.dogpile.com/index.html
http://www.euroseek.net/page?ifl=uk
http://www.excite.com/
http://www.go2net.com/search.html
http://www.google.com/
http://www.hotbot.com/
http://www.infoseek.com/
http://www.inktomi.com/
http://www.lycos.com/
http://www.northernlight.com/
http://www.snap.com/
http://www.stpt.com/
http://www.yahoo.com/
```

COPYRIGHT