



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'asa.1p' command

\$ man asa.1p

ASA(1P) POSIX Programmer's Manual ASA(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

asa ? interpret carriage-control characters

SYNOPSIS

asa [file...]

DESCRIPTION

The asa utility shall write its input files to standard output, mapping carriage-control characters from the text files to line-printer control sequences in an implementation-defined manner.

The first character of every line shall be removed from the input, and the following actions are performed.

If the character removed is:

<space> The rest of the line is output without change.

0 A <newline> is output, then the rest of the input line.

1 One or more implementation-defined characters that causes an advance to the next page shall be output, followed by the rest of the input line.

+ The <newline> of the previous line shall be replaced with one

or more implementation-defined characters that causes printing to return to column position 1, followed by the rest of the input line. If the '+' is the first character in the input, it shall be equivalent to <space>.

The action of the `asa` utility is unspecified upon encountering any character other than those listed above as the first character in a line.

OPTIONS

None.

OPERANDS

file A pathname of a text file used for input. If no file operands are specified, the standard input shall be used.

STDIN

The standard input shall be used if no file operands are specified, and shall be used if a file operand is '-' and the implementation treats the '-' as meaning standard input. Otherwise, the standard input shall not be used. See the INPUT FILES section.

INPUT FILES

The input files shall be text files.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of `asa`:

LANG Provide a default value for the internationalization variables that are unset or null. (See the Base Definitions volume of POSIX.1?2017, Section 8.2, Internationalization Variables for the precedence of internationalization variables used to determine the values of locale categories.)

LC_ALL If set to a non-empty string value, override the values of all the other internationalization variables.

LC_CTYPE Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments and input files).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error.

NLSPATH Determine the location of message catalogs for the processing of LC_MESSAGES.

ASYNCHRONOUS EVENTS

Default.

STDOUT

The standard output shall be the text from the input file modified as described in the DESCRIPTION section.

STDERR

None.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

0 All input files were output successfully.

>0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

None.

EXAMPLES

1. The following command:

```
asa file
```

permits the viewing of file (created by a program using FORTRAN-style carriage-control characters) on a terminal.

2. The following command:

```
a.out | asa | lp
```

formats the FORTRAN output of a.out and directs it to the printer.

RATIONALE

The `asa` utility is needed to map "standard" FORTRAN 77 output into a form acceptable to contemporary printers. Usually, `asa` is used to pipe data to the `lp` utility; see `lp`.

This utility is generally used only by FORTRAN programs. The standard developers decided to retain `asa` to avoid breaking the historical large base of FORTRAN applications that put carriage-control characters in their output files. There is no requirement that a system have a FORTRAN compiler in order to run applications that need `asa`.

Historical implementations have used an ASCII `<form-feed>` in response to a `1` and an ASCII `<carriage-return>` in response to a `+`. It is suggested that implementations treat characters other than `0`, `1`, and `+` as `<space>` in the absence of any compelling reason to do otherwise. However, the action is listed here as "unspecified", permitting an implementation to provide extensions to access fast multiple-line slewing and channel seeking in a non-portable manner.

FUTURE DIRECTIONS

None.

SEE ALSO

`fort77`, `lp`

The Base Definitions volume of POSIX.1-2017, Chapter 8, Environment Variables

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are

most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html .

IEEE/The Open Group

2017

ASA(1P)