NAME

spr − special print command

SYNOPSIS

spr [**−f** file] [**−u** dir] [**−h** header] [**−wn**] [**−ln**] [**−in**] [**−cn**] [**−d**] [**−t**] [**−b**] [**−v**] [**−s**] [+n] name ...

DESCRIPTION

Spr produces a listing of one or more files. Its special features include:

− incremental printing, i.e. printing only files which have changed since the last print.
− table of contents.
− control of heading and sub-heading.
− embedded control of listing format.

Normally, *spr* prints its *name* arguments or the files listed in the *file* argument. If neither of these arguments is present, *spr* prints its standard input.

The options have the following meanings:

−f *file* *File* must be a list of of files to be printed. Internally, *file* should contain a list of files, one per line. Since *spr* looks up file names relative to the directory from which it is run; the file names in *file* must be correct relative to the current working directory (see *pwd*(1)). Complete path names are safest.

−u *dir* *Dir* is the name of a file which will be used to record file status information. If you wish to do "incremental" listings, you must provide a *dir* argument. Assuming that a *dir* argument is given to *spr* and that file status information exists in *dir*, then only files which have been modified; or whose status has not been recorded in *dir* by a previous *spr* will be printed.

−h *header* Use *header* instead of the name of the file being printed as the default main heading.

−w*n* The output page width is taken to be *n* columns instead of the default (80).

−l*n* The output page length is considered to be *n* lines instead of the default (66).

−i*n* First line to be used for printing is line *n*. Default is line 0.

−c*n* First column to be used is column *n*. Default is column 0.

−d Causes table of contents to be printed.

−t Turns off top headings. Default is on.

−b Turns on bottom headings. Default is off.

−v Turns on both top and bottom headings. Overrides both **−t** and **−b** options.

−s Turns on subheadings. Default is off.

+*n* First file to print is *n*; file status information is gathered on all files but nothing is printed until the *n*th file is reached. The page number of the table of contents is 999, thus +999 will cause only the table of contents to be printed. A word about page numbers; *spr* prints page numbers in the form:

nnn−mmm

where *nnn* is the "page" or "file" number and *mmm* is the page within the "file".

name *Names* of files to be printed. These files are printed in addition to (and before) the files printed by the **−f** option.

NOTE: The **−d**, **−t**, **−b**, **−v**, and **−s** options are merely on/off flags in *spr*. That means that successive occurences of the option will turn the flag to its opposite state. For instance:

```
spr −d −d −d
```

will cause the state of the table of contents flag to change from off to on to off to on.

EXAMPLE

```
spr −f fillst −u dir −s −d extra −s extra2
```

Causes *extra* to be printed with subheadings, *extra2* to be printed without subheadings and then prints all files listed in *fillst* without subheadings. After all files have been printed, a table of contents will be printed which includes *extra*, *extra2* and all files in *fillst*. File status information will be maintained on *dir* for all files printed.

EMBEDDED CONTROL SEQUENCES

Lines of the form:

```
/*.x'par1'par2'*/
```

are commands to *spr*. The character, ('), represents a delimiter which may be any character except **<nl>** or blank. These lines are not listed in prints produced by *spr*; if it is desired to see these lines, use *pr*(1) or *cat*(1). Note that the format of these lines is such that they will be ignored by the C compiler. The following are the *spr* commands (i.e. values which *x* may assume):

s Create a subheading of the form:

par1: *par2*

The **s** command also causes a page eject. *Par1* is a character string, ≤ 10 characters (usually the name of a subroutine) and *par2* is a comment whose length is ≤ 40 characters.

t Same as **s** except that the subheading is not changed and no page eject is done. The purpose of **t** is to make entries for the table of contents.

n Same effect as **s** except that no page eject is done. That is, the subheading on the page that the **n** command occurs will change, but it will not cause a page eject. The most common use of the **n** command was immediately following the # line in C files; however, with current versions of the C compiler that kluge is no longer necessary and so the value of the **n** command has diminished.

i The **i** command is intended to be used to make subentries in the table of contents under subroutine entries. *Par1* is a ≤ 10 character string which will appear in the table of contents indented 3 spaces after the **s**, **t**, or **n** entry which it follows. The first letter of *par2* appears in the description column of the table of contents. For instance:

```
/*.i'aardvark'G'*/
```

Might be used to document the appearance of the Global variable *aardvark*.

m The **m** command overrides the current main heading − which is normally taken to be the name of the file being printed (see **−h** option to *spr*). In this case, *par1* is a ≤ 40 character string.

e Causes an immediate page eject.

Except for the **m** and **e** commands, all commands cause entries to be made in the table of contents (assuming one is printed).

TABLE OF CONTENTS

The table of contents consists of 2 parts. The first part contains 1 line per file printed giving the file number (see part 2 description below), name of the file and the date of the last modification of the file. Part 2 contains a listing of all **s**, **n** and **t** entries together with their subtending **i** entries sorted into alphabetic order. For each file printed, a default **s** entry is made where *par1* is the name of the file and *par2* is the full path name of the file (as given in the **−f** file or *name*). Each **s** entry line (or **n** or **t**) contains *par1*, the file number (see description of part 1), the page number on which the *spr* command occured and *par2*. Each **i** entry gives *par1* indented 3 spaces, the page number and the first character of *par2*.

DIAGNOSTICS

Presumably, all diagnostics are self−explanatory. However, the diagnostics:

> *par*: **param error**

and

> **come again ?**

are likely to be confusing. Both of these messages are an attempt to say that something was wrong with the parameters given when *spr* was invoked.

BUGS

The file names are limited to 40 characters.

Also, *spr* was originally written for use on output devices which have 80 columns per line and 66 lines per page. Some of its computations on the legality of the input values given to it assume that those limits are "absolute". For instance, if you change the first line to 5, you should change the page length to 61; otherwise, *spr* will add 5 to 66 and get 71 which is longer than it believes a page has any right to be. *Spr* should be fixed so that it does not assume anything about the output medium.