

NAME

`tabfunc` - convert table to functions for `rcalc`, etc.

SYNOPSIS

`tabfunc` [**`-i`**] func1 [func2 ..]

DESCRIPTION

Tabfunc reads a table of numbers from the standard input and converts it to an expression suitable for *icalc(1)*, *rcalc(1)* and their cousins. The input must consist of a M x N matrix of real numbers, with exactly one row per line. The number of columns must always be the same in each line, separated by whitespace and/or commas, with no missing values. The first column is always the independent variable, whose value indexes all of the other elements. This value does not need to be evenly spaced, but it must be either monotonically increasing or monotonically decreasing. (I.e. it cannot go up and then down, or down and then up.) Maximum input line width is 4096 characters and the maximum number of data rows is 1024. Input lines not beginning with a numerical value will be silently ignored.

The command-line arguments given to *tabfunc* are the names to be assigned to each column. *Tabfunc* then produces a single function for each column given. If there are some columns which should be skipped, the dummy name "0" may be given instead of a valid identifier. (It is not necessary to specify a dummy name for extra columns at the end of the matrix.)

The `-i` option causes *tabfunc* to produce a description that will interpolate values in between those given for the independent variable on the input.

EXAMPLE

To convert a small data table and feed it to `rcalc` for some calculation:

```
rcalc -e 'tabfunc f1 f2 < table.dat' -f com.cal
```

AUTHOR

Greg Ward

SEE ALSO

`cnt(1)`, `icalc(1)`, `neaten(1)`, `rcalc(1)`, `rlam(1)`, `total(1)`