

**tr**

**COLLABORATORS**

|               |                      |                |                  |
|---------------|----------------------|----------------|------------------|
|               | <i>TITLE :</i><br>tr |                |                  |
| <i>ACTION</i> | <i>NAME</i>          | <i>DATE</i>    | <i>SIGNATURE</i> |
| WRITTEN BY    |                      | August 3, 2022 |                  |

**REVISION HISTORY**

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
|        |      |             |      |

---

# Contents

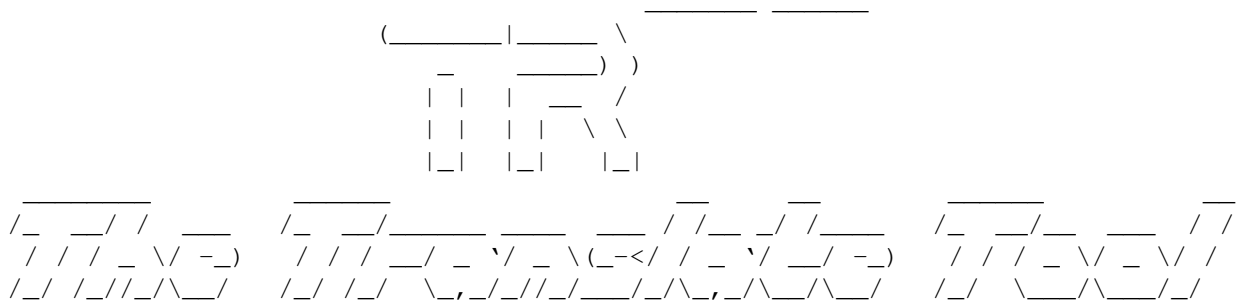
|          |                 |          |
|----------|-----------------|----------|
| <b>1</b> | <b>tr</b>       | <b>1</b> |
| 1.1      | main . . . . .  | 1        |
| 1.2      | usage . . . . . | 1        |
| 1.3      | dist . . . . .  | 4        |
| 1.4      | req . . . . .   | 4        |
| 1.5      | both . . . . .  | 4        |

---

# Chapter 1

## tr

### 1.1 main

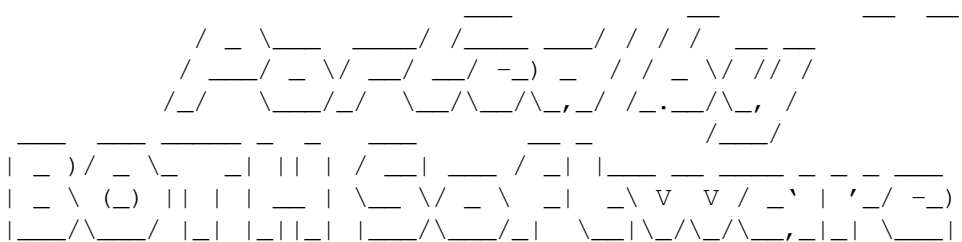


Usage

Requirements

Distribution

Authors



### 1.2 usage

Tr (Translate)

Tr is a UNIX tool that examines a file and changes characters within it. The version of TR included here is an Amiga port of the FreeBSD

source. The Man page for Tr is included, but is very cryptic.

No external libraries are required.

Tr can be used from the shell or from within a script. Redirection is supported.

The following document is the man page from the original GNU source.  
tr.man

General usage from within a DOS Script

Tr performs three basic functions on text files:

- d \* Deletes unwanted characters
- t \* Changes characters (Translate from one to another) (default)
- s \* Removes repeated characters (Squeeze repeated characters)

Format

Tr can be used in backticks or have it's output redirected. Here are two examples:

```
SETENV OUTFILE `tr "A-Z" "a-z" <infile`
```

```
tr >outfile "A-Z" "a-z" <infile
```

Both forms will take input from <infile> and convert all upper case letters to lower case. The result is written to <outfile>.

The examples perform a TRANSLATE operation. The first string "A-Z" is what Tr is looking for in the infile. The second "a-z" is what will be substituted in the outfile.

Examples of TRANSLATING characters

The -t switch is not needed for translation. In the following example we will change all occurrences of "K" to "C".

```
SETENV OUTFILE `tr "K" "C" <infile`
```

Special characters and punctuation may be specified by preceding them with a backspace. Non printing characters may be specified. Here is a list of these characters:

|      |   |
|------|---|
| \a   | Control-G   |
| \b   | Control-H   |
| \f   | Control-L   |
| \n   | Control-J   |
| \r   | Control-M   |
| \t   | Control-I   |
| \v   | Control-K   |
| \ooo | The character with the value given by ooo, which is |

1 to 3 octal digits.  
\ A backslash.

To change newlines to carriage returns, use this form:

```
SETENV OUTFILE `tr "/n" "/r" <infile`
```

#### DELETING CHARACTERS

The `-d` switch makes `Tr` remove selected characters from a file. The following example will remove all occurrences of the letter "X" from a file:

```
SETENV OUTFILE `tr -d "X" <infile`
```

#### SQUEEZING OUT REPEATED CHARACTERS

The `-s` switch causes `Tr` to delete repeated characters, leaving only one of the characters in its place. To demonstrate this, assume the infile is this:

```
"Thaaaaats All Folks!"
```

To remove the extra a's use this form:

```
SETENV OUTFILE `tr -s "a" <infile`  
ECHO $OUTFILE  
Thats All Folks!
```

This is fine if you know there are extra a's in the file. If you want to do a broad based squeeze, try this:

```
SETENV OUTFILE `tr -s ":alpha:" <infile`
```

This will squeeze out all repeated alpha characters. The `:alpha:` notation is a "character class". These classes are common groupings of character types. They save you a lot of typing. See the man page above for a listing of character classes.

#### DELETING AND SQUEEZING CHARACTERS

The `-d` and `-s` switches may be used together. `Tr` first deletes and then squeezes the result.

#### COMPLEMENT (the NOT function)

The `-c` switch replaces the characters specified with their complement. An example:

If you specify "A-Z" with the `-c` switch, everything BUT A-Z are specified. In Amiga terms, this is a NOT function.

The following example uses the Complement and Squeeze switches in combination. The line will clean up a typical UNIX man page, removing all tabs, non printing characters, escape codes, etc., leaving the file clean so it may be viewed properly.

---

It does a Squeeze on all characters not specified. An easy way to look at this operation is to put all the characters you want unchanged inside the string, and then do a compliment on it.

CLEANUP A MAN PAGE:

```
SETENV OUTFILE `tr -c -s "a-z A-Z 0-9 /n /r /space /punct" <man.page`
```

Once you have mastered the basics of Tr, refer to the man page above for more details.

### 1.3 dist

Tr was ported from the original FreeBSD source. Both Software assumes no liability for any damage the program may cause.

Tr may be freely copied and distributed.

Source files are included.

### 1.4 req

Tr can be used on any Amiga running OS 2.xx and up.  
The ixemul.lib is not required.

### 1.5 both

If you find any bugs or have any comments, please contact the Authors. Email messages will be answered immediately. Snail Mail letters will take longer since we never learned to read.

C & UNIX Code

-----  
Will Bow  
1531 Corinth Unit 1  
West Los Angeles, CA 90025  
(310) 478-4913  
InterNet:wbow@crl.com

Program Design, GUIs, Docs

-----  
Colin Thompson  
9606 Carroll Cyn. Rd. H9  
San Diego, CA 92126  
(619) 695-2181  
InterNet: colin@cts.com

-----