

**NAME**

**cat** — concatenate files

**SYNOPSIS**

**cat** [**-us**] [**-n**|-**b**] [**-TEv**] [**-teA**] [*file*]...

**DESCRIPTION**

Copies each *file* (standard input stream if "-", the default), in order, to the standard output stream. In case of an input error, the file is skipped but processing continues.

**OPTIONS**

<b>-u</b>	Don't buffer output. Input is never buffered.
<b>-s, --squeeze-blank</b>	Replace runs of empty input lines with a single one.
<b>-n, --number</b>	Number lines from <i>l</i> , separating them from the <i>file</i> contents with a tab.
<b>-b, --number-nonblank</b>	Likewise, but don't number empty lines.
<b>-T, --show-tabs</b>	Replace tab characters with " <b>^I</b> ".
<b>-E, --show-ends</b>	Prepend newlines with a dollar (" <b>\$</b> ").
<b>-v, --show-nonprinting</b>	Replace non-printable bytes ( $[0, 0x1F] \cup [0x7F, 0xFF] - \{0x09 \text{ (tab)}, 0x0A \text{ (newline)}\}$ ) with their corresponding " <b>^c</b> " and " <b>M-c</b> " escapes (see <b>STANDARDS, Escapes</b> ).
<b>-t</b>	<b>-Tv</b>
<b>-e</b>	<b>-Ev</b>
<b>-A, --show-all</b>	<b>-TEv</b>

**EXIT STATUS**

**1** if a *file* couldn't be opened or read.

**EXAMPLES**

```
$ echo echoed | cat
echoed

$ echo abcdefg | split -b3
$ printf : | cat xaa - xab - xac
abc:defg
$ cat xaa - xab - xac # teletype input bolded
abcA
A
^DdefB
B
^Dg

$ cat form
Groceries for February:

        Bananas 3.5kg   $4.51
        Kiwis    2kg    $3.19   Call Siegfried to explain short!
        Bread    $20.21

$ cat -sbte form
1 Groceries for February:$
$
2 ^IBananas^I3.5kg^I$4.51$
3 ^IKiwis^I2kg^I$3.19^ICall Siegfried to explain short!$
4 ^IBread^I^I$20.21$
```

**SEE ALSO**

cp(1), dd(1), head(1), pr(1), tail(1), ascii(7)

**STANDARDS**

Conforms to IEEE Std 1003.1-2024 (“POSIX.1”); **-u** is the only flag specified by the standard. **-nb**, **-s**, **-tev** compatible with 4BSD. Remaining flags (**-TEA**) compatible with the GNU system.

**Escapes**

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xA	0xB	0xC	0xD	0xE	0xF
0x00	^@	^A	^B	^C	^D	^E	^F	^G	^H	^I		^K	^L	^M	^N	^O
0x10	^P	^Q	^R	^S	^T	^U	^V	^W	^X	^Y	^Z	^[	^\	^]	^^	^_
0x70																^?

Above **0x80**, the table repeats but with **M-** prepended (i.e. **0xC1** is **M-A** and **0x8A** — **M-^J**).

**HISTORY**

Appears in the first edition of the UNIX Programmer’s Manual as **cat(I)**:

NAME `cat` -- concatenate and print

SYNOPSIS `cat file1 ...`

With the standard input stream used if no *files*, as present-day.

Version 3 AT&T UNIX adds the **-**-as-standard-input-stream behaviour. This can safely be considered fully-formed.

Version 7 AT&T UNIX sees a rewrite in C with `<stdio.h>`, and **-u** disabling buffering of the standard output stream (considering it feeds `getc()` directly into `putchar()` until end-of-file, this is a significant penalty, but is, regardless, most likely decidedly slower than the previous implementation, which mapped full-block `read(II)s` onto `write(II)s`). It also explicitly checks for, skips, and warns about one of the input files also being bound to the standard output stream.

3BSD uses a larger, *1024*-byte `BUFSIZ`, and hence output buffer.

4BSD introduces **-nb**, **-s**, and **-tve** — all as today.

4.3BSD sees a fast `read(2)/write(2)` path if no flags were specified.

Programmer’s Workbench (PWB/UNIX), instead, sees **-s**, silencing errors, and a re-write in terms of `read(2)/write(2)`. Oddly, AT&T System III UNIX inherits this **-s** into a Version 7 AT&T UNIX base.

AT&T System V Release 2 UNIX also sees **-tve**, except **-te** require **-v** to activate, **-t** also controls **0x12** (form feed), and **M-^J** (newline + high bit) is **M- $\$$ <newline>**. The former can be attributed to implementing according to the 4BSD manpage, which erroneously implies the **-v** requirement. The latter only to low QoI. Without **-v**, a block-wise `read(2)/write(2)` copy is performed.