NAME od – octal dump

SYNOPSIS
/usr/bin/od [–**bcCDdFfOoSsvXx**] [–] [*file*] [*offset_string*]
/usr/bin/od [–**bcCDdFfOoSsvXx**] [–**A** *address_base*] [–**j** *skip*]
[–**N** *count*] [–**t** *type_string*] . . . [–] [*file* . . .]

/usr/xpg4/bin/od [–**bcCDdFfOoSsvXx**] [–] [*file*] [*offset_string*]
/usr/xpg4/bin/od [–**bcCDdFfOoSsvXx**] [–**A** *address_base*] [–**j** *skip*]
[–**N** *count*] [–**t** *type_string* . . .] [–] [*file* . . .]

AVAILABILITY
/usr/bin/od SUNWtoo

/usr/xpg4/bin/od SUNWxcu4

DESCRIPTION The **od** command copies sequentially each input file to standard output and transforming the input data according to the output types specified by the **-t** or –**bcCDdFfOoSsvXx** options. If no output type is specified, the default output is as if –**t o2** had been specified. Multiple types can be specified by using multiple –**bcCDdFfOoSstvXx** options. Output lines are written for each type specified in the order in which the types are specified. If no *file* is specified, the standard input is used. The [*offset_string*] operand is mutually exclusive from the –**A**, –**j**, –**N**, and –**t** options. For the purposes of this description, the following terms are used:

word	refers to a 16-bit unit, independent of the word size of the machine
long word	refers to a 32-bit unit
double long word	refers to a 64-bit unit.

OPTIONS The following options are supported:

–**A** *address_base*
Specify the input offset base. The *address_base* option-argument must be a character. The characters **d**, **o** and **x** specify that the offset base will be written in decimal, octal or hexadecimal, respectively. The character **n** specifies that the offset will not be written. Unless –**A n** is specified, the output line will be preceded by the input offset, cumulative across input files, of the next byte to be written. In addition, the offset of the byte following the last byte written will be displayed after all the input data has been processed. Without the –**A** *address_base* option and the [*offset_string*] operand, the input offset base is displayed in octal.

–**b** Interpret bytes in octal. This is equivalent to –**t o1**.

/usr/bin/od
–**c** Display single-byte characters. Certain non-graphic characters appear as C-language escapes:

null	**\0**
backspace	**\b**
form-feed	**\f**

new-line	**\n**
return	**\r**
tab	**\t**

others appear as 3-digit octal numbers. For example:

```
echo "hello world" | od -c
0000000  h  e  l  l  o     w  o  r  l  d \n
0000014
```

/usr/xpg4/bin/od

−c Interpret bytes as single-byte or multibyte characters according to the current setting of the **LC_CTYPE** locale category. Printable multibyte characters are written in the area corresponding to the first byte of the character; the two character sequence ** is written in the area corresponding to each remaining byte in the character, as an indication that the character is continued. Non-graphic characters appear the same as they would using the −**C** option.

−C Interpret bytes as single-byte or multibyte characters according to the current setting of the **LC_CTYPE** locale category. Printable multibyte characters are written in the area corresponding to the first byte of the character; two character sequence ** are written in the area corresponding to each remaining byte in the character, as an indication that the character is continued. Certain non-graphic characters appear as C escapes:

null	**\0**
backspace	**\b**
formfeed	**\f**
newline	**\n**
return	**\r**
tab	**\t**

Other non-printable characters appear as one three-digit octal number for each byte in the character.

−d Interpret words in unsigned decimal. This is equivalent to −**t u2**.

−D Interpret long words in unsigned decimal. This is equivalent to −**t u4**.

−f Interpret long words in floating point. This is equivalent to −**t f4**.

−F Interpret double long words in extended precision. This is equivalent to −**t f8**.

−j *skip* Jump over *skip* bytes from the beginning of the input. The **od** command will read or seek past the first *skip* bytes in the concatenated input files. If the combined input is not at least *skip* bytes long, the **od** command will write a diagnostic message to standard error and exit with a non-zero exit status.

By default, the *skip* option-argument is interpreted as a decimal number. With a leading **0x** or **0X**, the offset is interpreted as a hexadecimal number; otherwise, with a leading **0**, the offset will be interpreted as an octal number. Appending the character **b**, **k** or **m** to offset will cause it to be interpreted as a multiple of **512**, **1024** or **1 048 576** bytes, respectively. If the *skip* number is hexadecimal, any appended **b** is considered to be the final hexadecimal digit. The address is displayed starting at **0000000**, and its base is not implied by the base of the *skip* option-argument.

–N *count*
Format no more than *count* bytes of input. By default, *count* is interpreted as a decimal number. With a leading **0x** or **0X**, *count* is interpreted as a hexadecimal number; otherwise, with a leading **0**, it is interpreted as an octal number. If *count* bytes of input (after successfully skipping, if **–j** *skip* is specified) are not available, it will not be considered an error; the **od** command will format the input that is available. The base of the address displayed is not implied by the base of the *count* option-argument.

–o Interpret words in octal. This is equivalent to **–t o2**.

–O Interpret long words in unsigned octal. This is equivalent to **–t o4**.

–s Interpret words in signed decimal. This is equivalent to **–t d2**.

–S Interpret long words in signed decimal. This is equivalent to **–t d4**.

–t *type_string*
Specify one or more output types. The *type_string* option-argument must be a string specifying the types to be used when writing the input data. The string must consist of the type specification characters:

a *Named character*. Interpret bytes as named characters. Only the least significant seven bits of each byte will be used for this type specification. Bytes with the values listed in the following table will be written using the corresponding names for those characters.

Named Characters in od

Value	Name	Value	Name	Value	Name	Value	Name
\000	**nul**	\001	**soh**	\002	**stx**	\003	**etx**
\004	**eot**	\005	**enq**	\006	**ack**	\007	**bel**
\010	**bs**	\011	**ht**	\012	**lf**	\013	**vt**
\014	**ff**	\015	**cr**	\016	**so**	\017	**si**
\020	**dle**	\021	**dc1**	\022	**dc2**	\023	**dc3**
\024	**dc4**	\025	**nak**	\026	**syn**	\027	**etb**
\030	**can**	\031	**em**	\032	**sub**	\033	**esc**
\034	**fs**	\035	**gs**	\036	**rs**	\037	**us**
\040	**sp**	\177	**del**				

c *Character*. Interpret bytes as single-byte or multibyte characters specified by the current setting of the **LC_CTYPE** locale category. Printable multibyte characters are written in the area corresponding to the first byte of the character; the two character sequence ** is written in the area corresponding to each remaining byte in the character, as an indication that the character is continued. Certain non-graphic characters appear as C escapes: **\0**, **\a**, **\b**, **\f**, **\n**, **\r**, **\t**, **\v.** Other non-printable characters appear as one three-digit octal number for each byte in the character.

The type specification characters **d**, **f**, **o**, **u** and **x** can be followed by an optional unsigned decimal integer that specifies the number of bytes to be transformed by each instance of the output type.

f *Floating point*. Can be followed by an optional **F**, **D** or **L** indicating that the conversion should be applied to an item of type **float**, **double** or **long double**, respectively.

d, **o**, **u** and **x**
Signed decimal, *octal*, *unsigned decimal*, and *hexadecimal*, respectively. Can be followed by an optional **C**, **S**, **I** or **L** indicating that the conversion should be applied to an item of type **char**, **short**, **int** or **long**, respectively.

Multiple types can be concatenated within the same *type_string* and multiple **–t** options can be specified. Output lines are written for each type specified in the order in which the type specification characters are specified.

–v Show all input data (verbose). Without the **–v** option, all groups of output lines that would be identical to the immediately preceding output line (except for byte offsets), will be replaced with a line containing only an asterisk (*).

–x Interpret words in hex. This is equivalent to **–t x2**.

–X Interpret long words in hex. This is equivalent to **–t x4**.

OPERANDS

The following operands are supported:

– Use the standard input in addition to any files specified. When this operand is not given, the standard input is used only if no *file* operands are specified.

/usr/bin/od

file A path name of a file to be read. If no *file* operands are specified, the standard input will be used. If there are no more than two operands, none of the –**A**, –**j**, –**N** or –**t** options is specified, and *any* of the following are true:

1. the first character of the last operand is a plus sign (+)
2. the first character of the second operand is numeric
3. the first character of the second operand is **x** and the second character of the second operand is a lower-case hexadecimal character or digit
4. the second operand is named "**x**"
5. the second operand is named "**.**"

then the corresponding operand is assumed to be an offset operand rather than a file operand.

Without the –**N** count option, the display continues until an end-of-file is reached.

/usr/xpg4/bin/od

file — Same as **/usr/bin/od**, except only one of the first two conditions must be true.

/usr/bin/od

[+] [0] *offset* **[.] [b | B]**
[+] [0] [*offset*] **[.]**
[+][0x | x][*offset*]
[+][0x | x] *offset* **[B]** — The *offset_string* operand specifies the byte offset in the file where dumping is to commence. The offset is interpreted in octal bytes by default. If *offset* begins with "**0**", it is interpreted in octal. If *offset* begins with "**x**" or "**0x**", it is interpreted in hexadecimal and any appended "**b**" is considered to be the final hexadecimal digit. If "." is appended, the offset is interpreted in decimal. If "**b**" or "**B**" is appended, the offset is interpreted in units of **512** bytes. If the *file* argument is omitted, the *offset* argument must be preceded by a plus sign (**+**). The address is displayed starting at the given offset. The radix of the address will be the same as the radix of the offset, if specified, otherwise it will be octal. Decimal overrides octal, and it is an error to specify both hexadecimal and decimal conversions in the same offset operand.

/usr/xpg4/bin/od

[+] [0] *offset* **[.] [b | B]**
+ [*offset***] [.]**
[+][0x][*offset*]
[+][0x] *offset* **[B]**
+x [*offset*]
+x*offset* **[B]** — Description of *offset_string* is the same as for **/usr/bin/od**.

ENVIRONMENT

See **environ**(5) for descriptions of the following environment variables that affect the execution of **od**: **LC_CTYPE**, **LC_MESSAGES**, **LC_NUMERIC**, and **NLSPATH**.

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

SEE ALSO

sed(1), **environ**(5)