NAME

cksum – write file checksums and sizes

SYNOPSIS

cksum [*file* . . .]

AVAILABILITY

SUNWcsu

DESCRIPTION

The **cksum** command calculates and writes to standard output a cyclic redundancy check (CRC) for each input file, and also writes to standard output the number of octets in each file.

For each file processed successfully, **cksum** will write in the following format:

> **"%u %d %s\n"** <*checksum*>, <*# of octets*>, <*path name*>

If no *file* operand was specified, the path name and its leading space will be omitted.

The CRC used is based on the polynomial used for CRC error checking in the referenced Ethernet standard.

The encoding for the CRC checksum is defined by the generating polynomial:

$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

Mathematically, the CRC value corresponding to a given file is defined by the following procedure:

1. The n bits to be evaluated are considered to be the coefficients of a mod 2 polynomial $M(x)$ of degree $n-1$. These n bits are the bits from the file, with the most significant bit being the most significant bit of the first octet of the file and the last bit being the least significant bit of the last octet, padded with zero bits (if necessary) to achieve an integral number of octets, followed by one or more octets representing the length of the file as a binary value, least significant octet first. The smallest number of octets capable of representing this integer is used.
2. $M(x)$ is multiplied by x^{32} (that is, shifted left 32 bits) and divided by $G(x)$ using mod 2 division, producing a remainder $R(x)$ of degree ≤ 31.
3. The coefficients of $R(x)$ are considered to be a 32-bit sequence.
4. The bit sequence is complemented and the result is the CRC.

OPERANDS

The following operand is supported:

file A path name of a file to be checked. If no *file* operands are specified, the standard input is used.

USAGE

The **cksum** command is typically used to quickly compare a suspect file against a trusted version of the same, such as to ensure that files transmitted over noisy media arrive intact. However, this comparison cannot be considered cryptographically secure. The chances of a damaged file producing the same CRC as the original are astronomically small; deliberate deception is difficult, but probably not impossible.

Although input files to **cksum** can be any type, the results need not be what would be expected on character special device files. Since this document does not specify the block size used when doing input, checksums of character special files need not process all of the data in those files.

The algorithm is expressed in terms of a bitstream divided into octets. If a file is transmitted between two systems and undergoes any data transformation (such as moving 8-bit characters into 9-bit bytes or changing "Little Endian" byte ordering to "Big Endian"), identical CRC values cannot be expected. Implementations performing such transformations may extend **cksum** to handle such situations.

ENVIRONMENT

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

EXIT STATUS

The following exit values are returned:

0 All files were processed successfully.

>0 An error occurred.

SEE ALSO

sum(1), **environ**(5)