NAME

uuencode, uudecode – encode a binary file, or decode its encoded representation

SYNOPSIS

uuencode [*source-file*] *decode_pathname*

uudecode [–**p**] [*encoded-file*]

AVAILABILITY

SUNWesu

DESCRIPTION

uuencode

uuencode converts a binary file into an encoded representation that can be sent using **mail**(1). It encodes the contents of *source-file*, or the standard input if no *source-file* argument is given. The *decode_pathname* argument is required. The *decode_pathname* is included in the encoded file's header as the name of the file into which **uudecode** is to place the binary (decoded) data. **uuencode** also includes the permission modes of *source-file,* (except **setuid**, **setgid**, and sticky-bits), so that *decode_pathname* is recreated with those same permission modes.

uudecode

uudecode reads an *encoded-file*, strips off any leading and trailing lines added by mailer programs, and recreates the original binary data with the filename and the mode specified in the header.

The encoded file is an ordinary portable character set text file; it can be edited by any text editor. It is best only to change the mode or *decode_pathname* in the header to avoid corrupting the decoded binary.

OPTIONS

uudecode

–**p** decode *encoded-file* and send it to standard output. This allows **uudecode** to be used in a pipeline.

OPERANDS

uuencode

The following operands are supported by **uuencode**:

decode_pathname
The pathname of the file into which the **uudecode** utility will place the decoded file. If there are characters in *decode_pathname* that are not in the portable filename character set the results are unspecified.

source-file
A pathname of the file to be encoded.

uudecode

The following operand is supported by **uudecode**:

encoded-file
The pathname of a file containing the output of **uuencode**.

ENVIRONMENT

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

OUTPUT

stdout

The standard output is a text file (encoded in the character set of the current locale) that begins with the line:

"begin∆%s∆%s\n", <*mode*>, *decode_pathname*

and ends with the line:

end\n

In both cases, the lines have no preceding or trailing blank characters.

The algorithm that is used for lines in between **begin** and **end** takes three octets as input and writes four characters of output by splitting the input at six-bit intervals into four octets, containing data in the lower six bits only. These octets are converted to characters by adding a value of 0x20 to each octet, so that each octet is in the range 0x20–0x5f, and then it is assumed to represent a printable character. It then will be translated into the corresponding character codes for the codeset in use in the current locale. (For example, the octet 0x41, representing **A ,** would be translated to **A** in the current codeset, such as 0xc1 if it were EBCDIC.)

Where the bits of two octets are combined, the least significant bits of the first octet are shifted left and combined with the most significant bits of the second octet shifted right. Thus the three octets A, B, C are converted into the four octets:

```
0x20 + (( A >> 2                    ) & 0x3F)
0x20 + (((A << 4) | ((B >> 4) & 0xF)) & 0x3F)
0x20 + (((B << 2) | ((C >> 6) & 0x3)) & 0x3F)
0x20 + (( C                         ) & 0x3F)
```

These octets are then translated into the local character set.

Each encoded line contains a length character, equal to the number of characters to be decoded plus 0x20 translated to the local character set as described above, followed by the encoded characters. The maximum number of octets to be encoded on each line is 45.

EXIT STATUS

The following exit values are returned:

0 Successful completion.

>0 An error occurred.

SEE ALSO

mail(1), **mailx**(1), **uucp**(1C), **uux**(1C)

NOTES

The encoded file's size is expanded by 35% (3 bytes become 4, plus control information), causing it to take longer to transmit than the equivalent binary.

The user on the remote system who is invoking **uudecode** (typically **uucp**) must have write permission on the file specified in the *decode_pathname*.

If you **uuencode** then **uudecode** a file in the same directory, you will overwrite the original file.