

uudecode

Decode an UU-encoded string

Description

```
uudecode( dest[], const source[], maxLength=sizeof dest )
```

Since the UU-encoding scheme is for binary data, the decoded data is always “packed”. The data is unlikely to be a string (the zero-terminator may not be present, or it may be in the middle of the data).

A buffer may be decoded “in-place”; the destination size is always smaller than the source size. Endian issues (for multi-byte values in the data stream) are not handled.

Binary data is encoded in chunks of 45 bytes. To assemble these chunks into a complete stream, function memcpy allows you to concatenate buffers at bytealigned boundaries

NOTE: This is a rarely needed command for ezeio programming.

Parameters

dest	The buffer that will hold the decoded data
source	The uu-encoded source buffer
maxLength (optional)	The size of dest in cells. If the length of dest would exceed maxLength cells, the result is truncated.

Return value

The number of bytes stored in dest.

Example usage

From:
<https://doc.eze.io/> - **ezeio documentation**

Permanent link:
<https://doc.eze.io/ezeio2/scriptref/uudecode>

Last update: **2019-09-02 21:45**

