

NAME

alloc – core allocator

SYNOPSIS

char *alloc(size)

free(ptr)

char *ptr;

DESCRIPTION

Alloc and *free* provide a simple general-purpose core management package. *Alloc* is given a size in bytes; it returns a pointer to an area at least that size which is even and hence can hold an object of any type. The argument to *free* is a pointer to an area previously allocated by *alloc*; this space is made available for further allocation.

Needless to say, grave disorder will result if the space assigned by *alloc* is overrun or if some random number is handed to *free*.

The routine uses a first-fit algorithm which coalesces blocks being freed with other blocks already free. It calls *sbrk* (see *break (II)*) to get more core from the system when there is no suitable space already free, and writes “Out of space” on the standard output, then exists, if that fails.

The external variable *slop* (which is 2 if not set) is a number such that if *n* bytes are requested, and if the first free block of size at least *n* is no larger than *n+slop*, then the whole block will be allocated instead of being split up. Larger values of *slop* tend to reduce fragmentation at the expense of unused space in the allocated blocks.

DIAGNOSTICS

“Out of space” if it needs core and can’t get it.

BUGS