

Pablo Iván Martín Enríquez

C dynamic memory allocation refers to performing manual memory management for dynamic memory allocation in the C programming language via a group of functions in the C standard library, namely malloc, realloc, calloc and free.

The C dynamic memory allocation functions are defined in `stdlib.h` header (`cstdlib` header in C++)

- malloc: allocates the specified number of bytes
- realloc: increases or decreases the size of the specified block of memory, moving it if necessary
- calloc: allocates the specified number of bytes and initializes them to zero
- free: libera el bloque de memoria especificada de nuevo al sistema

malloc returns a void pointer to the allocated space, or NULL if there is insufficient memory available. To return a pointer to a type other than void, use a type cast on the return value. The storage space pointed to by the return value is guaranteed to be suitably aligned for storage of any type of object that has an alignment requirement less than or equal to that of the fundamental alignment.

realloc returns a void pointer to the reallocated (and possibly moved) memory block. If there is not enough available memory to expand the block to the given size, the original block is left unchanged, and NULL is returned. If size is zero, then the block pointed to by memblock is freed; the return value is NULL, and memblock is left pointing at a freed block.

calloc returns a pointer to the allocated space. The storage space pointed to by the return value is guaranteed to be suitably aligned for storage of any type of object. To get a pointer to a type other than void, use a type cast on the return value.