bYTE ORDER MANUPULATION

The first group of functions, whose names begin with b (for byte), The second group of functions, whose names begin with mem (for memory), are from the ANSI C standard and are provided with any system that supports an ANSI C library.

#include <strings.h>

void bzero(void *_dest_, size_t _nbytes_);

void bcopy(const void *_src_, void *_dest_, size_t _nbytes_);

int bcmp(const void *_ptr1_, const void *_ptr2_, size_t _nbytes_);

bzero sets the specified number of bytes to 0 in the destination. We often use this function to initialize a socket address structure to 0. bcopy moves the specified number of bytes from the source to the destination. bcmp compares two arbitrary byte strings. The return value is zero if the two byte strings are identical; otherwise, it is nonzero.

#include <string.h>

void *memset(void *_dest_, int _c_, size_t _len_);

void *memcpy(void *_dest_, const void *_src_, size_t _nbytes_);

int memcmp(const void *_ptr1_, const void *_ptr2_, size_t _nbytes_);

memset sets the specified number of bytes to the value _c_ in the destination.

memcpy is similar to bcopy, but the order of the two pointer arguments is swapped.