Documentazione Programmazione Socket

//*Structures for handling internet addresses*
struct sockaddr_in {// ...
      short sin_family; // *e.g. AF_INET, AF_INET6*
      unsigned short sin_port; // *e.g. htons(3490)*
      struct in_addr sin_addr; // *see struct in_addr* };

struct in_addr{ unsigned long s_addr; //*e.g. INADDR_ANY* };

//*Structure for handling host names*
struct hostent{// ...
      char *h_name; //*The real canonical host name.*
      int h_addrtype; //*The result's address type, e.g. AF_INET*
      int length; //*The length of the addresses in bytes, which is 4 for IP (version 4) addresses.*
      h_addr; //*An IP address for this host.* };

int socket(int domain, int type, int protocol); //*Allocate a socket descriptor*

//*Accept an incoming connection on a listening socket*
int accept(int sockfd, struct sockaddr *addr, socklen_t *addrlen);

//*Associate a socket with an IP address and port number*
int bind(int sockfd, struct sockaddr *my_addr, socklen_t addrlen);

int connect(int sockfd, const struct sockaddr *addr, socklen_t addrlen); //*Connect - initiate a connection on a socket*

int close(int sockfd); //*Close a socket descriptor*

struct hostent *gethostbyname(const char *name); //*Get an IP address for a hostname*

//*Convert multi-byte integer types from host byte order to network byte order*
uint32_t htonl(uint32_t hostlong);
uint16_t htons(uint16_t hostshort);
uint32_t ntohl(uint32_t netlong);
uint16_t ntohs(uint16_t netshort);

//*Convert IP addresses to human-readable form and back*
const char *inet_ntop(int af, const void *src, char *dst, socklen_t size);
int inet_pton(int af, const char *src, void *dst);

int listen(int sockfd, int queuelength); //*Tell a socket to listen for incoming connections*

ssize_t recv(int sockfd, void *buf, size_t len, int flags); //*Receive data on a socket*
ssize_t send(int sockfd, const void *buf, size_t len, int flags); //*Send data out over a socket*

//*Receive data on a socket*
ssize_t recvfrom(int sockfd, void *buf, size_t len, int flags, struct sockaddr *src_addr, socklen_t *addrlen);
//*Send data out over a socket*
ssize_t sendto(int sockfd, const void *buf, size_t len, int flags, const struct sockaddr *dest_addr, socklen_t addrlen);

ssize_t read (int fd, void *buf, size_t count); //*Read data from a stream socket*
ssize_t write (int fd, const void *buf, size_t count); //*Write data on a stream socket*

UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Utility Functions

void bzero(void *s, size_t n); //*Set the first n bytes of the area starting at s to zero*

void bcopy(const void *src, void *dest, size_t n); // *Copy n bytes from src to dest.*

int strcmp(const char *s1, const char *s2); // *Compare the two strings s1 and s2.*
int strncmp(const char *s1, const char *s2, size_t n); // *Compare the first n byte of s1 and s2.*