

```

//=====file=getname.c =====Pseudo Code=====
//= A program to get the host name for a given IP address
=
//=====
==
//= Notes:
=
//= 1) This program conditionally compiles for Winsock and BSD sockets.
=
//= Set the initial #define to WIN or BSD as appropriate.
=
//= 2) This program assumes command line entry of the IP address.
=
//-----
-=
//= Example execution:
=
//=
=
//= =
//= Build: bcc32 getname.c or cl getname.c wsock32.lib for Winsock
=
//= gcc getname.c -lsocket -lnsl for BSD
=
//-----
-=
//= Execute: getname IP_address
=
//-----
-=
//-----
-=
#define WIN // WIN for Winsock and BSD for BSD sockets

//----- Include files -----
-
#include <stdio.h> // Needed for printf()
#include <stdlib.h> // Needed for exit()
#include <string.h> // Needed for memcpy() and strcpy()

#ifdef WIN
#include <windows.h> // Needed for all Winsock stuff
#endif
#ifdef BSD
#include <sys/types.h> // Needed for system defined identifiers.
#include <netinet/in.h> // Needed for internet address structure.
#include <arpa/inet.h> // Needed for "inet_ntoa".
#include <sys/socket.h> // Needed for socket(), bind(), etc...
#include <fcntl.h>
#include <netdb.h> // Need for gethostbyaddr().
#endif

//===== Main
program=====
void main(int argc, char *argv[])
{

```

```

#ifdef WIN
    WORD wVersionRequested = MAKEWORD(1,1);          // Stuff for WSA functions
    WSADATA wsaData;                                // Stuff for WSA functions
#endif

    struct hostent *host;                            // Structure for gethostbyaddr()
    struct in_addr *myaddr;                           // Structure for Internetaddress
    char ip_address[256];                             // String for IP address

    if (argc != 2)
    {
        printf("need host's IP address as commmand line arguments \n");
        printf("usage is 'getname IP_address' \n");
        exit(1);
    }

#ifdef WIN
    // This stuff initializes winsock
    WSASStartup(wVersionRequested, &wsaData);
#endif

    // Copy IP address into ip_address
    strcpy(ip_address, argv[1]);

    myaddr=(struct in_addr*)malloc(sizeof(struct in_addr));
    myaddr->s_addr=inet_addr(ip_address) ;

    // Do a gethostbyaddr() to get a pointer to struct host
    printf("Looking for host name for for '%s'... \n", ip_address);
    host = gethostbyaddr((char *) myaddr, 4 ,AF_INET);

    // Output host name if host found
    if (host == NULL)
        printf(" Host name for '%s' could not be found \n",ip_address);
    else
        printf(" Host name for '%s' is '%s' \n",ip_address, host->h_name);

#ifdef WIN
    // This stuff cleans-up winsock
    WSACleanup();
#endif

}

```