

```

/* Svolgimento Compito 1 - 21/12/05 */

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <string.h>
#include <dirent.h>
#include <fcntl.h>
#define LENGTH_NAME 20

int main(int argc, char **argv)
{
    struct hostent *host;
    struct sockaddr_in clientaddr, servaddr;
    int sd, len, ris;
    char nome_stanza[LENGTH_NAME];

    /* CONTROLLO ARGOMENTI ----- */
    if(argc!=3)
    {
        printf("Error:%s serverAddress serverPort\n", argv[0]);
        exit(1);
    }

    /* PREPARAZIONE INDIRIZZO CLIENT E SERVER ----- */
    memset((char *)&clientaddr, 0, sizeof(struct sockaddr_in));
    clientaddr.sin_family = AF_INET;
    clientaddr.sin_addr.s_addr = INADDR_ANY;
    clientaddr.sin_port = 0;

    memset((char *)&servaddr, 0, sizeof(struct sockaddr_in));
    servaddr.sin_family = AF_INET;
    host = gethostbyname (argv[1]);
    if (host == NULL)
    {
        printf("%s not found in /etc/hosts\n", argv[1]);
        exit(2);
    }
    else
    {
        servaddr.sin_addr.s_addr=((struct in_addr *)(host->h_addr))->s_addr;
        // Controllare anche numero porta!!
        servaddr.sin_port = htons(atoi(argv[2]));
    }

    printf("Client avviato\n");

    /* CREAZIONE SOCKET ----- */
    sd=socket(AF_INET, SOCK_DGRAM, 0);
    if(sd<0) {perror("apertura socket"); exit(3);}
    printf("Creata la socket sd=%d\n", sd);
    /* BIND SOCKET, a una porta scelta dal sistema ----- */
    if(bind(sd,(struct sockaddr *) &clientaddr, sizeof(clientaddr))<0)
    {perror("bind socket "); exit(1);}
    printf("Client: bind socket ok, alla porta %i\n", clientaddr.sin_port);

    /* CORPO DEL CLIENT: */

```

```

/* ciclo di accettazione di richieste di conteggio ----- */

printf("Richieste di sospensione stanze fino alla fine del file di input\n");
printf("Qualsiasi tasto per procedere, EOF per terminare\n");
printf("Nome della stanza: ");

while (gets(nome_stanza))
{
    printf("Richiesta sospensione stanza %s \n", nome_stanza);

    /* invio richiesta */
    len=sizeof(servaddr);
    if (sendto(sd, nome_stanza, (strlen(nome_stanza)+1), 0, (struct sockaddr
*)&servaddr, len)<0)
    {
        perror("scrittura socket");
        printf("Nome stanza da sospendere: ");
        continue; // se questo invio fallisce il client torna all'inizio del ciclo
    }

    /* ricezione del risultato */
    printf("Attesa del risultato...\n");
    if (recvfrom(sd, &ris, sizeof(ris), 0, (struct sockaddr *)&servaddr, &len)<0)
    {
        perror("recvfrom");
        printf("Nome della stanza da sospendere: ");
        continue; // se questa ricezione fallisce il client torna all'inizio del ciclo
    }

    ris=ntohl(ris);
    if(ris==0) printf("Sospensione effettuata con successo!\n");
    else if(ris==-1) printf("Problemi ad effettuare la sospensione\n");
    else printf("Problemi, risposta non attesa!\n");

    printf("Nome della stanza da sospendere: ");

} // while

printf("\nClient: termino...\n");
close(sd);

exit(0);
}

```