

Examen Febrero 2019 resuelto.pdf



Yassine_bhk



Fundamentos de Software de Comunicaciones



2º Grado en Ingeniería de las Tecnologías de Telecomunicación



Escuela Técnica Superior de Ingeniería de Telecomunicación
Universidad de Málaga



**Invita a otros estudiantes, crea contenido y
gana los premios que te alegrarán el verano**

participa aquí



Hasta el 15/06/2023

“No necesitas besos, si no volcanes de fuego en tu boca”.
¿De quién es la frase?

- a) William Shakespeare
- b) Federico García Lorca
- c) Nicholas Leister.

Culpa Mía

prime

VER AHORA

```
/* CLIENTE */
#include <stdio.h>
#include <unistd.h>
#include <errno.h>
#include <signal.h>
#include <stdint.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <string.h>
#include <sys/time.h>
#include <fcntl.h>

#define PUERTO 4950

ssize_t readn(int fd, void* p, size_t max){
    ssize_t leidos = 0;
    size_t a_leer = max;
    char *m = p;
    size_t r;
    do{
        errno = 0;
        r = read(fd,m + leidos,a_leer);
        if(r > 0){
            leidos += r;
            a_leer -= r;
        }
    }while(((a_leer != 0) && (r > 0)) || (errno == EINTR));
    if(r < 0){
        return -1;
    }else{
        return leidos;
    }
}

ssize_t writen(int fd, void* p, size_t max){
    ssize_t escritos = 0;
    size_t a_escribir = max;
    char *m = p;
    size_t w;
    do{
```

WUOLAH

```

        errno = 0;
        w = write(fd, m + escritos, a_escribir);
        if(w > 0){
            escritos += w;
            a_escribir -= w;
        }
    }while(((w > 0) && (a_escribir != 0)) || (errno == EINTR));
    if(w < 0){
        return -1;
    }else{
        return escritos;
    }
}

int fin = 0;

void manejadora (int s) {

    fin = 1;
}

int main (int argc, char * argv []) {

    if (argc < 2) {

        printf ("Introduzca un argumento porfavor");
        exit (1);
    }

    int sd = socket (AF_INET, SOCK_STREAM, 0);

    if (sd < 0) {

        perror ("Socket");
        exit (1);
    }

    signal (SIGINT, manejadora);

    struct sockaddr_in vinculo;
    memset (&vinculo, 0, sizeof (vinculo));

```




No mires debajo de la cama.

THE BOOGEYMAN

YA EN CINES

COMPRAR ENTRADAS



©2023 20th Century Studios.

```

vinculo.sin_addr.s_addr = inet_addr (argv [1]);
vinculo.sin_family = PF_INET;
vinculo.sin_port = htons (PUERTO);

if (connect (sd, (struct sockaddr *) & vinculo, sizeof (vinculo)) < 0) {

    perror ("connect");
    close (sd);
    exit (1);
}

while (!fin) {

    printf ("Porfavor, escriba la temperatura...\n");
    int16_t temperatura;
    char buffer [512];
    int leidos = read (0, buffer, 512);
    buffer [leidos - 1] = '\0';
    temperatura = atoi (buffer);

    printf ("temperatura escrita: %d\n", temperatura);

    if (leidos < 0) {

        perror ("Leyendo");
        close (sd);
        exit (1);
    }

    if (temperatura < -260 || temperatura > 153) {

        printf ("Temperatura no valida\n");
    }

    else {

        int16_t temperatura_n = htons (temperatura);
        int escritos = writen (sd, &temperatura_n, 2);
        printf ("Enviado con éxito...\n");

        if (escritos < 0) {

```

```
        perror ("EScribiendo");  
        close (sd);  
        exit (1);  
    }  
}  
  
}  
  
close (sd);  
  
}
```

“No necesitas besos, si no volcanes de fuego en tu boca”.
¿De quién es la frase?

- a) William Shakespeare
- b) Federico García Lorca
- c) Nicholas Leister.

Culpa Mía

prime

VER AHORA

```
/* SERVIDOR*/
#include <stdio.h>
#include <unistd.h>
#include <errno.h>
#include <signal.h>
#include <stdint.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <string.h>
#include <sys/time.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/wait.h>

#define PUERTO 4950

ssize_t readn(int fd, void* p, size_t max){
    ssize_t leidos = 0;
    size_t a_leer = max;
    char *m = p;
    size_t r;
    do{
        errno = 0;
        r = read(fd,m + leidos,a_leer);
        if(r > 0){
            leidos += r;
            a_leer -= r;
        }
    }while(((a_leer != 0) && (r > 0)) || (errno == EINTR));
    if(r < 0){
        return -1;
    }else{
        return leidos;
    }
}

ssize_t writen(int fd, void* p, size_t max){
    ssize_t escritos = 0;
    size_t a_escribir = max;
```

WUOLAH


```

char *m = p;
size_t w;
do{
    errno = 0;
    w = write(fd, m + escritos, a_escribir);
    if(w > 0){
        escritos += w;
        a_escribir -= w;
    }
}while(((w > 0) && (a_escribir != 0)) || (errno == EINTR));
if(w < 0){
    return -1;
}else{
    return escritos;
}

}

int maximo (int a, int b) {

    if (a > b) {

        return a;
    }

    else {

        return b;
    }
}

int llegada = 0;

void manejadora (int s) {

    if (llegada == 0) {

        printf ("Timeout");
        signal (s, manejadora);
    }

}

```



```

void manejadora2 (int s) {

    printf ("Warning");
    signal (s, manejadora);

}

int fin = 0;

int main (int argc, char * argv []) {

    int sd = socket (AF_INET, SOCK_STREAM, 0);
    signal (SIGALRM, manejadora);

    if (sd < 0) {

        perror ("Socket");
        exit (1);
    }

    struct sockaddr_in vinculo;
    memset (&vinculo, 0, sizeof (vinculo));
    vinculo.sin_addr.s_addr = INADDR_ANY;
    vinculo.sin_family = PF_INET;
    vinculo.sin_port = htons (PUERTO);

    if (bind (sd, (struct sockaddr *) & vinculo, sizeof (vinculo)) < 0) {

        perror ("bind");
        close (sd);
        exit (1);
    }

    if (listen (sd, 10) < 0) {

        perror ("Listen");
        close (sd);
        exit (1);
    }

    int pid_hijo = fork ();

```

```

    if (pid_hijo < 0) {

        perror ("fork");
        close (sd);
        exit (1);
    }

    if (pid_hijo == 0) {

        signal (SIGUSR1, manejadora2);

        while (1) {

            pause ();
        }

        exit (0);
    }

    struct sockaddr_in info_cliente1, info_cliente2, info_cliente3;
    socklen_t tam1 = sizeof (info_cliente1);
    socklen_t tam2 = sizeof (info_cliente2);
    socklen_t tam3 = sizeof (info_cliente3);

    int csd1 = accept (sd, (struct sockaddr *) &info_cliente1, &tam1);
    int csd2 = accept (sd, (struct sockaddr *) &info_cliente2, &tam2);
    int csd3 = accept (sd, (struct sockaddr *) &info_cliente3, &tam3);

    if (csd1 < 0 || csd2 < 0 || csd3 < 0) {

        perror ("accept");
        close (sd);
        exit (1);
    }

    int16_t temperatura1;
    int16_t temperatura2;
    int16_t temperatura3;

    fd_set conjunto;

```

“No necesitas besos, si no volcanes de fuego en tu boca”.
¿De quién es la frase?

- a) William Shakespeare
- b) Federico García Lorca
- c) Nicholas Leister.

Culpa Mia

prime

VER AHORA

```
FD_SET (csd1, &conjunto);
FD_SET (csd2, &conjunto);
FD_SET (csd3, &conjunto);

do {

    int max_fd = maximo (csd1, csd2);
    max_fd = maximo (max_fd, csd3);
    //alarm (5);
    int r = select (max_fd + 1, &conjunto, 0, 0, 0);

    if (r < 0 && errno != EINTR) {

        perror ("Select");
        exit (1);
    }

    if (FD_ISSET (csd1, &conjunto)) {

        llegada++;
        int16_t temperatura_n;
        int leidos = readn (csd1, &temperatura_n, 2);

        if (leidos < 0) {

            perror ("Leyendo");
            close (csd1);
            close (csd2);
            close (csd3);
            close (sd);
            exit (1);
        }

        else if (leidos == 0) {

            fin = 1;
            FD_CLR (csd1, &conjunto);
        }

        else if (leidos > 0) {
```

WUOLAH

```

    temperatur1 = ntohs (temperatura_n);
    printf ("Temperatura recibida: %d\n", temperatur1);

    if (temperatur1 < -260 || temperatur1 > 153) {

        printf ("Warning : valor incorrecto.\n");

    }

}

} //FD_ISSET

if (FD_ISSET (csd2, &conjunto)) {

    llegada++;
    int16_t temperatura_n;
    int leidos = readn (csd2, &temperatura_n, 2);

    if (leidos < 0) {

        perror ("Leyendo");
        close (csd1);
        close (csd2);
        close (csd3);
        close (sd);
        exit (1);
    }

    else if (leidos == 0) {

        fin = 1;
        FD_CLR (csd2, &conjunto);
    }

    else if (leidos > 0) {

        temperatura2 = ntohs (temperatura_n);
        printf ("Temperatura recibida: %d\n", temperatura2);

        if (temperatura2 < -260 || temperatura2 > 153) {

            printf ("Warning : valor incorrecto.\n");

        }
    }
}

```

```

    }

    } //FD_ISSET

if (FD_ISSET (csd3, &conjunto)) {

    llegada++;
    int16_t temperatura_n;
    int leidos = readn (csd3, &temperatura_n, 2);

    if (leidos < 0) {

        perror ("Leyendo");
        close (csd1);
        close (csd2);
        close (csd3);
        close (sd);
        exit (1);
    }

    else if (leidos == 0) {

        fin = 1;
        FD_CLR (csd2, &conjunto);
    }

    else if (leidos > 0) {

        temperatura3 = ntohs (temperatura_n);
        printf ("Temperatura recibida: %d\n", temperatura3);

        if (temperatura3 < -260 || temperatura3 > 153) {

            printf ("Warning : valor incorrecto.\n");
        }
    }
} //FD_ISSET

if (abs (temperatura2 - temperatura3) >= 5) {

    printf ("Warning: Lectura no válida 1.\n");
    kill (pid_hijo, SIGUSR1);
}

```



```

        else if (abs (temperatura1 - temperatura2) >= 5) {

            printf ("Warning: Lectura no válida 2.\n");
            kill (pid_hijo, SIGUSR1);
        }

        else if (abs (temperatura1 - temperatura3) >= 5) {

            printf ("Warning: Lectura no válida 3.\n");
            kill (pid_hijo, SIGUSR1);
        }

    } while (!fin);

    kill (pid_hijo, SIGTERM);

    close (sd);

}

```

“No necesitas besos, si no volcanes de fuego en tu boca”.
¿De quién es la frase?

- a) William Shakespeare
- b) Federico García Lorca
- c) Nicholas Leister.

Culpa Mía

prime

VER AHORA

```
/* EJERCICIO 1*/
#include <stdio.h>
#include <unistd.h>
#include <errno.h>
#include <signal.h>
#include <stdint.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <string.h>
#include <sys/time.h>

#include <fcntl.h>

#define EST_LEYENDO 0
#define EST_CRITICO 1

#define EV_TIMEOUT 0
#define EV_ERROR1 1
#define EV_ERROR2 2
#define EV_WARNING 3

ssize_t readn (int fd, void * buffer, size_t n) {

    char * p = (char *) buffer;
    int intentando_leer = n;
    int total_leidos = 0;
    int leidos;

    do {

        errno = 0;
        leidos = read (fd, p + total_leidos, intentando_leer);

        if (leidos > 0) {

            total_leidos += leidos;
            intentando_leer -= leidos;
        }
    } while (leidos > 0 && total_leidos < n || errno == EINTR);
```

WUOLAH

```

    if (leidos < 0) {

        return leidos;
    }

    else {

        return total_leidos;
    }
}

ssize_t writen (int fd, void * buffer, size_t n) {

    char * p = (char *) buffer;
    int intentando_leer = n;
    int total_leidos = 0;
    int leidos;

    do {

        errno = 0;
        leidos = write (fd, p + total_leidos, intentando_leer);

        if (leidos > 0) {

            total_leidos += leidos;
            intentando_leer -= leidos;
        }
    } while (leidos > 0 && total_leidos < n || errno == EINTR);

    if (leidos < 0) {

        return leidos;
    }

    else {

        return total_leidos;
    }
}

```

```

}

int fin = 0;
int errores = 0;
int fd_pipe [2];
int fd;

void manejadora (int s) {

    uint32_t evento = EV_TIMEOUT;
    int e = writen (fd_pipe [1], &evento, 4);

    if (e < 0) {

        perror ("Escribeinefo");
        exit (1);
    }

    signal (s, manejadora);
}

int maximo (int a, int b) {

    if (a > b) {

        return a;
    }

    else {

        return b;
    }
}

int espera_evento () {

    uint32_t evento;

    fd_set conjunto;
    FD_ZERO (&conjunto);
    FD_SET (fd, &conjunto);
    FD_SET (fd_pipe [0], &conjunto);

```

```

int max = maximo(fd, fd_pipe [0]);

int resultado = select (max + 1, &conjunto, 0, 0, 0);

if (resultado < 0) {

    if (errno == EINTR) {

    }

    else {

        perror ("Select");
        close (fd);
        close (fd_pipe [0]);
        exit (1);

    }

}

if (FD_ISSET (fd, &conjunto)) {

    char letra [2];
    int leidos = readn (fd, letra, 2);
    letra [leidos - 1] = '\0';
    printf ("Letra leida: %s ", letra);

    if (leidos < 0) {

        perror ("Leyendo");
        close (fd);
        close (fd_pipe [0]);
        exit (1);

    }

    if (leidos == 0) {

        FD_CLR (fd, &conjunto);

    }
}

```


“No necesitas besos, si no volcanes de fuego en tu boca”.
¿De quién es la frase?

- a) William Shakespeare
- b) Federico García Lorca
- c) Nicholas Leister.

Culpa Mia

prime

VER AHORA

```
if (strcmp (letra, "r") == 0 || strcmp (letra, "R") == 0) {

    printf ("-> Es igual a r o R\n");

    if (errores == 0) {

        evento = EV_ERROR1;
        errores++;

    }

    else {

        evento = EV_ERROR2;
        errores = 0;
    }
} //comprobando si es una r

else { //if (letra == 'w' || letra == 'W') {

    printf ("-> No es igual a r o R\n");
    evento = EV_WARNING;printf ("warning: %d\n", evento);
} //comprobando si es una w

} //fd_isset

if (FD_ISSET (fd_pipe [0], &conjunto)) {

    evento = EV_TIMEOUT;

} //fd_isset

return evento;

} //espera evento

int main () {

    fd = open ("fifo", O_RDONLY);
    int resultado = pipe (fd_pipe);

    if (resultado < 0) {
```

WUOLAH

```

        perror ("pipe");
        exit (1);
    }

    signal (SIGALRM, manejadora);

    if (fd == -1) {

        perror ("Open");
        exit (1);
    }

    int estado = EST_LEYENDO;

    struct itimerval timer;
    struct timeval t = {3, 500000};

    while (!fin) {

        int evento = espera_evento ();
        printf ("Espera_evento: %d\n", evento);
        printf ("Estado: %d\n", estado);

        if (evento == -1) {

            perror ("Evento no encontrado");
            exit (1);
        }

        switch (estado)
        {
        case EST_LEYENDO:

            switch (evento)
            {
            case EV_ERROR1:

                break;

            case EV_ERROR2:

                estado = EST_CRITICO;

```

```

        printf ("Cambiando de estado...\n");
        timer.it_interval = t;
        timer.it_value = t;
        setitimer (ITIMER_REAL, &timer, 0);

        break;

case EV_TIMEOUT:

    printf ("Evento no espeado timeout");
    exit (1);

    break;

case EV_WARNING:

    printf ("WARNING...\n");

    break;

default:

    printf ("Evento no esperado.");
    exit (1);

    break;
}

break;

case EST_CRITICO:

    switch (evento)
    {

        case EV_ERROR1:

            printf ("Terminando maquina");
            close (fd);
            fin = 1;

            break;

```

```

    case EV_ERROR2:

        printf ("Evento no esperado 2 error");
        exit (1);

        break;

    case EV_TIMEOUT:

        estado = EST_LEYENDO;

        break;

    case EV_WARNING:

        printf ("WARNING...\n");

        break;

    default:

        printf ("Evento no esperadodefault.");
        exit (1);

        break;
}

break;

break;

default:
    break;
}
}
}

```