



Instituto Politécnico Nacional  
Escuela Superior de Cómputo



### **TAREA 3**

**Sincronizar los procesos del programa de tuberías sin nombre**

**Grupo: 2CM8**

**Alumno: Morales Blas David Israel**

**Asignatura: Sistemas Operativos**

**Profesora: Ana Belem Juárez Méndez**

**Fecha: 17-Marzo-2020**

## Descripción:

En la tarea mostrada a continuación se busca sincronizar procesos mediante el uso de tuberías sin nombre, con ello el programa mostrado a continuación usará 2 tuberías con las cuales el proceso padre e hijo se comunicarán, este programa se detendrá hasta que el padre escriba "EXIT".

## Código:

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <stdlib.h>
#include <string.h>
#include <sys/wait.h>
void limpiarCad(char *);
int main(){
    int pid,exitt;
    int tuberia[2],tuberia2[2];
    char mensaje[250],respuesta[250];

    //Creando la tuberia
    if(pipe(tuberia)== -1 && pipe(tuberia2)== -1){
        perror("\n Error en la tuberia \n");
        exit(-1);
    }
    pid=fork();
    if(pid==-1){
        perror("\nError al crear al hijo en fork\n");
        exit(-1);
    }else if(pid==0) {
        //HIJO
        exitt=1;
        while(exitt){
            read(tuberia[0], mensaje,sizeof(mensaje));
            printf("\nHIJO- El valor leído es: %s\n", mensaje);
            strcpy(respuesta,"HIJO: respuesta ");
            limpiarCad(respuesta);

            write(tuberia2[1],respuesta,strlen(respuesta)+1);
            limpiarCad(respuesta);
            exitt=strcmp(mensaje, "EXIT\n");
        }
        close(tuberia[0]);
        close(tuberia[1]);
        close(tuberia[0]);
        close(tuberia[1]);
        exit(0);
    }
```

```

    }else{
        //PADRE
        exitt=1;
        while(exitt){
            printf("\nPADRE: Da un valor: ");
            fgets(mensaje, sizeof(mensaje),stdin);
            write(tuberia[1], mensaje,strlen(mensaje)+1);
            read(tuberia2[0],respuesta,sizeof(respuesta));
            limpiarCad(respuesta);
            printf("\nPadre- Respuesta del hijo: %s\n", respuesta);
            exitt=strcmp(mensaje,"EXIT\n");
            sleep(1);
        }
        wait(NULL);
        close (tuberia[0]);
        close (tuberia[1]);
        close (tuberia2[0]);
        close (tuberia2[1]);
    }
return 0;
}
void limpiarCad(char *cadena) {
    char *p = strchr(cadena, '\n');
    if (p) {
        *p = '\0';
    }
}
}

```

#### CAPTURAS DE PANTALLA:

Se compila y luego se ejecuta el programa, el padre solicita un valor, el hijo lo muestra y posterior el hijo dará su respuesta como se muestra en la IMAGEN 1.1.

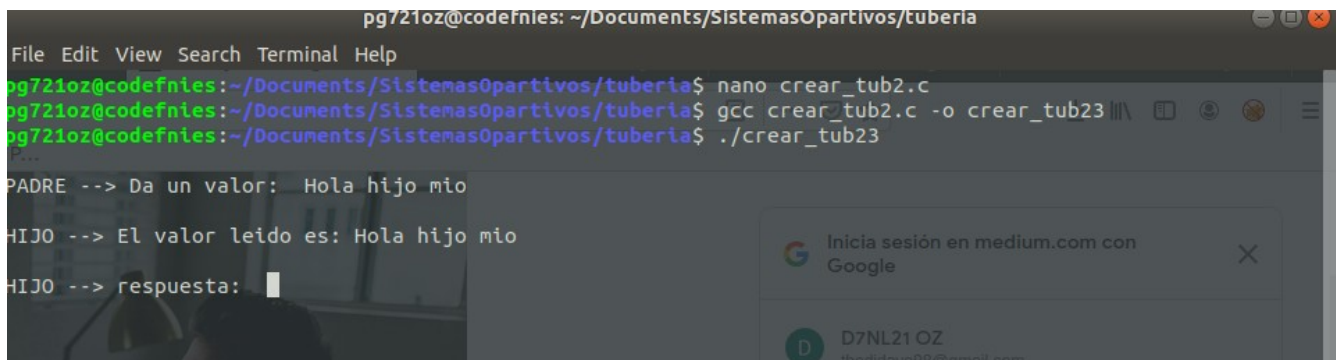


IMAGEN 1.1. Compilacion y ejecución del programa.

Ahora se mostrará en la IMAGEN 1.2 la conversación completa y como finaliza cuando el padre escribe "EXIT"

```
pg721oz@codefnies: ~/Documents/SistemasOpartivos/tuberia
File Edit View Search Terminal Help
pg721oz@codefnies:~/Documents/SistemasOpartivos/tuberia$ nano crear_tub2.c
pg721oz@codefnies:~/Documents/SistemasOpartivos/tuberia$ gcc crear_tub2.c -o crear_tub23
pg721oz@codefnies:~/Documents/SistemasOpartivos/tuberia$ ./crear_tub23

PADRE --> Da un valor:  Hola hijo mio
HIJO --> El valor leído es: Hola hijo mio
HIJO --> respuesta:  Hola papa
Padre --> Respuesta del hijo:  Hola papa
PADRE --> Da un valor:  El coronavirus esta fuerte
HIJO --> El valor leído es: El coronavirus esta fuerte
HIJO --> respuesta:  Si apa muy fuerte, no quieren que me titule :(
Padre --> Respuesta del hijo:  Si apa muy fuerte, no quieren que me titule :(
PADRE --> Da un valor:  Bueno hijo nos vemos
HIJO --> El valor leído es: Bueno hijo nos vemos
HIJO --> respuesta:  Si papa chao
Padre --> Respuesta del hijo:  Si papa chao
PADRE --> Da un valor:  EXIT
HIJO --> El valor leído es: EXIT
HIJO --> respuesta:  Adios !
Padre --> Respuesta del hijo:  Adios !
pg721oz@codefnies:~/Documents/SistemasOpartivos/tuberia$
```

IMAGEN 1.2. Visualización del programa completo en ejecución hasta salida con “EXIT” .

## CONCLUSIONES:

Como se puede observar es posible comunicar procesos mediante tuberías, esto al igual se puede usar mediante señales, el uso de tuberías sin nombre es sencillo de aplicar, ya que sólo se necesita entender que una tubería tiene una entrada y una salida, al igual es necesario comprender como poder escribir y leer de la misma, con esta práctica me quedó claro el como hacerlo.