

Técnicas Digitales III

Trabajo práctico: Socket

1. Analice los códigos de los programas servidor.c y cliente.c ¿Cuál es el objetivo de los programas?.

```
Compile el programa servidor.c    gcc -o servidor servidor.c
Compile el programa cliente.c      gcc -o cliente cliente.c
```

```
Ejecute servidor                  ./servidor
```

```
En otra consola ejecute cliente  ./cliente
```

Observe la salida por consola.

2. Termine los programas del ejercicio anterior. Con los programas del ejercicio 1:

```
Ejecute servidor                  ./servidor
```

```
En otra consola ejecute cliente  ./cliente
- Envíe datos desde el cliente al servidor .
```

```
En otra consola ejecute cliente  ./cliente
- Envíe datos desde el cliente al servidor . ¿Qué sucede?.
```

-Termine el primer proceso cliente y vea que sucede.

```
En otra consola ejecute cliente  ./clienteu
```

3. Termine todos los programas del ejercicio anterior. Con los programas del ejercicio 1:

```
Ejecute servidor                  ./servidor
```

```
En otra consola ejecute cliente  ./cliente
```

```
En otra consola ejecute cliente  ./cliente
```

```
En otra consola ejecute cliente  ./cliente
```

-¿Qué sucede cuando el tercer cliente intenta conectarse?

4. Analice los códigos de los programas `servidoru` y `clienteu` ¿Cuál es el objetivo de los programas?. ¿Qué diferencia encuentra con los programas `servidor` y `cliente` del ejercicio 1?

```
Compile el programa servidor.c      gcc -o servidoru servidoru.c
Compile el programa cliente.c       gcc -o clienteu clienteu.c
```

```
Ejecute servidor                    ./servidoru
```

```
En otra consola ejecute cliente     ./clienteu
- Envíe datos desde el cliente al servidor .
```

```
En otra consola ejecute cliente     ./clienteu
- Envíe datos desde el cliente al servidor .
```

```
En otra consola ejecute cliente     ./clienteu
- Envíe datos desde el cliente al servidor .
```

```
En otra consola ejecute cliente     ./clienteu
- Envíe datos desde el cliente al servidor .
```

- ¿Qué sucede?.

5. Modificar el programa `servidor` del ejercicio 1 para poder atender a varios clientes en simultáneo.

a- Para solucionar el problema utilice procesos hijos.

b- Para solucionar el problema utilice hilos. Recuerde incluir bibliotecas de hilos al compilar.

```
gcc -o servidor1 servidor1.c -lpthread
```