

Técnicas Digitales III

Trabajo práctico: Tuberías

Tuberías en C

1. Analice la estructura típica de implementación de tuberías. Compile y ejecute el programa pipe_01.c.

```
Compile el programa      gcc -o pipe_01 pipe_01.c
Ejecute                  ./pipe_01
```

2. Los procesos padre e hijo se comunican por una tubería. Compile y ejecute pipe_02.c. Comente todas las líneas indicando qué operación realiza cada una. ¿Qué problema observa en el programa? ¿Qué solución propone?
3. Los procesos padre e hijos se comunican por una tubería. Compile y ejecute pipe_03.c. Comente todas las líneas indicando qué operación realiza cada una. ¿Qué problema observa en el programa? ¿Qué solución propone?
4. Los procesos padre e hijo se comunican por una tubería. Compile y ejecute pipe_04.c. Comente todas las líneas indicando qué operación realiza cada una. ¿Qué problema observa en el programa? ¿Qué solución propone? Ejecute,

```
$ echo $?
```

para evaluar si el programa terminó bien (0) o con problemas.

Tuberías en consola (BASH shell)

5. Escriba un conjunto de comandos para buscar en todo el sistema de archivos, directorios o archivos que contengan la palabra network. Puede usar los comandos ls y grep, por ejemplo. Luego, copie estos comandos en un script para BASH (mi_script.sh) y ejecútelo desde consola.
6. Redirección de la salida estándar. Al usar el operador '>' en consola se puede redireccionar la salida de la consola a un archivo de texto. Ejecute los siguientes comandos en consola:

```
$ ls / | head -3 | tail -1 > myoutput
$ cat my_output
```

¿Qué operación realizan los comandos head, tail y cat? ¿Cuál es el contenido del archivo my_output?

¿Qué hace el operador '>>' ? ¿Qué diferencia presenta respecto al operador '>' ?

Tuberías en Python

7. Analice el script provisto `pipe_01.py`. Explique qué hace cada línea del script agregando un comentario por línea.
8. El objetivo de este ejercicio es calcular la distancia total recorrida por un drone usando la fórmula de Haversine, con la posición geográfica del drone como dato de entrada.

Cree un script en Python que realice las siguientes tareas:

- a. El proceso padre lee la posición del drone provista en el archivo `gps_data.csv`.
- b. El proceso padre transmite la posición por una tubería a un proceso hijo.
- c. El proceso hijo recibe este dato y calcula la distancia recorrida entre dos puntos con la fórmula de Haversine (https://en.wikipedia.org/wiki/Haversine_formula).
- d. El proceso hijo imprime por consola la distancia discreta de cada tramo y al final el total de la distancia recorrida.