

Sistemas Operativos

Práctica 2

1. Procesos.

Considere el programa elaborado en la práctica 1 en la que se gestiona información de tiempos de viaje. Para la práctica 2, es necesario diseñar dos programas cliente-servidor, que permitan acceder a los registros almacenados en disco.

1.1. Servidor.

El programa servidor deberá gestionar el archivo suministrado por Uber, que almacena los tiempos de viaje. A su vez, el servidor podrá recibir peticiones de clientes a través de la red, para realizar cada una de las solicitudes especificadas previamente: Ingresar origen, Ingresar destino, Ingresar hora, Buscar tiempo de viaje medio.

Al entrar en ejecución, el servidor no mostrará ningún menú ni mensaje alguno. El servidor se limitará a gestionar las operaciones con los clientes y un archivo *log* donde se registrarán las operaciones que se estén realizando. El formato *log* de cada operación es el siguiente:

```
[Fecha YYYYMMDDTHHMMSS] Cliente [IP] [búsqueda - origen - destino]
```

1.2. Cliente.

El programa cliente deberá mostrar un menú igual al de la práctica 1:

Bienvenido

1. Ingresar origen
2. Ingresar destino
3. Ingresar hora
4. Buscar tiempo de viaje medio
5. Salir

Cuando se digite una opción, esta deberá ser enviada al servidor a fin de iniciar su gestión. El formato de los comandos es libre y deberá especificarse en el documento de informe de elaboración.

1.3. Consideraciones.

- Hacer uso de punteros y de memoria dinámica (`malloc()` - `free()`). La cantidad de memoria usada por el proceso servidor no deberá superar 1MB, por lo que la información de los tiempos de viaje debe residir en disco.
- Por cada opción ejecutada, siempre se debe dar un mensaje de confirmación y solicitar **cualquier** tecla para continuar, antes de volver al menú principal.
- El número máximo de clientes que se pueden conectar al servidor es de 32.
- Entrega: Archivos fuentes **p2-server.c** **p2-client.c** **Makefile** y archivo **LEEME**. Todo dentro de una carpeta con los nombres que aparecen en el correo para cada integrante. Pej: **capedrazab-capedrazab**. Incluya más archivos fuente si lo desea. Los programas principales y ejecutables deberán llamarse igual y como se indica.

1.4. Calificación.

Se tendrán en cuenta los siguientes aspectos para la evaluación:

- Funcionamiento de los programas. Servidor: 30 % Cliente: 30 %
- Código limpio. 20 % (modular, tabulaciones, comentarios - básicos, declaración de constantes, etc.)
- Documentos: manual de uso y especificaciones 10 %, informe de elaboración (funciones, diagrama de comunicaciones, diagrama de bloques) 10 %. Se tiene en cuenta redacción y ortografía de los documentos.