

## **Tercer Proyecto**

Un problema que presentan los diferentes sistemas de sincronización de archivos (como DropBox, LiveDrive y SugarSync) es que solo funcionan conectados al Internet. Desde antes existían aplicaciones sencillas que permitían sincronizar máquinas en una red sin conectarse a la Web (por ejemplo, una laptop y una computadora de escritorio).

En este proyecto se pretende desarrollar una pequeña herramienta que permita la sincronización de un directorio entre dos máquinas.

### **Programa sincronizador**

El programa sincronizador se debe ejecutar en dos máquinas al mismo tiempo. En la primer máquina el programa recibirá como parámetro únicamente el nombre del directorio a sincronizar y quedará esperando conexiones en el puerto 8889. En la otra máquina se debe pasar como parámetros el nombre del directorio local a sincronizar y el IP de la primer máquina.

Al iniciarse el programa deberá identificar los archivos en su directorio local, almacenando su tamaño, hora y fecha de la última actualización. Debe comparar dicha información con un listado de su corrida anterior, y determinar cuáles archivos han cambiado. Si algún archivo ha cambiado, se eliminó, o se creó, se debe enviar esa información a la otra máquina para que realice los cambios adecuados. Dentro de dichos cambios está solicitar el archivo actualizado a la otra máquina.

- Una vez que se realiza la sincronización, los programas que corren en las máquinas deben terminar. La siguiente vez que se desee sincronizar se debe volver a ejecutar el programa en ambas máquinas.
- Tome en cuenta que los relojes de las máquinas involucradas pueden estar desincronizados. Para determinar cuáles archivos son mas recientes se debe realizar un cálculo de diferencias entre las horas de ambos relojes.
- Note que NO se deben escribir dos programas diferentes. Es el mismo programa que actúa como cliente o servidor dependiendo de los parámetros que se le pasen.

- Debe utilizar sockets para que se comuniquen los programas. Además, debe definir un formato y protocolo adecuado para enviar y recibir las solicitudes de archivos.
- Debe tomar en cuenta la posibilidad que un archivo haya sido modificado en ambas máquinas, provocando el caso de “copias en conflicto”. Aquí lo mejor es dejar las dos copias pero con nombres ligeramente diferentes.

## Documentación

Se deberá generar una documentación formal, en formato pdf, en donde se describan las diferentes etapas del desarrollo del proyecto, las decisiones de diseño que se tomaron, los mecanismos de programación utilizados, y los resultados de las diferentes pruebas al programa.

Dicha documentación deberá incluir al menos las siguientes secciones:

- Introducción
- Descripción del problema (este enunciado)
- Definición de estructuras de datos
- Descripción detallada y explicación de los componentes principales del servidor/cliente:
- Manejo de sockets
- Manejo de solicitudes
- Rutina de comparación
- Descripción del protocolos y formatos
- Análisis de resultados de pruebas
- Conclusiones sobre rendimiento y correctitud

## Forma de evaluación

- Programa (100 %)
- Manipulación sockets (20%)
- Manejo de solicitudes (20%)
- Rutina de comparación (20%)
- Mecanismo de bloqueo (20%)
- Transferencia correcta (20%)

Aclaración importante: La documentación es parte integral de cualquier programa. Si no se incluye en el documento final la documentación específica de alguno de los elementos indicados anteriormente, se asignará una nota de 0 en dicho rubro completo. Es decir, si no aparece en la documentación es prácticamente como si no lo hubiese realizado.

## **Consideraciones generales**

- El proyecto debe ser ejecutado en grupos de a lo más dos personas. Se puede desarrollar en forma individual pero bajo ninguna circunstancia se aceptarán grupos de tres o más personas.
- No se permite la copia de código entre grupos de estudiantes, tampoco es permitido utilizar clases o librerías adicionales (desarrolladas por terceros) para simplificar el manejo de las conexiones, cache, o copia de archivos.
- El proyecto debe ser resuelto mediante lenguaje C, sockets, y ambiente Unix.
- Se deberá crear un archivo zip que incluya: el código fuente de todos los programas desarrollados, binarios generados, y documentación en formato pdf.