I.T.C.R.

Departamento de Computación
Prof: Ing. Erika Marín Schumann.

Sistemas Operativos

File System
I Semestre del 2015

Definición general

Este proyecto tiene como finalidad implementar un File System. Para esto se implementaran las funciones del File System que se desglosan posteriormente. La idea es que se tenga una serie de comandos y el sistema responda a cada uno de ellos. La interfaz con el usuario tiene que ser gráfica. Con cada comando que se introduce (de la manera en que les parezca más usable) el sistema debe ejecutar la instrucción, tanto en lo que utilicen en memoria para el manejo del File System como en un archivo que representará el disco del file System que estamos haciendo. Pueden utilizar las estructuras de datos que ustedes diseñen para mantener la información del sistema en memoria. Sin embargo, en disco debe haber un archivo del tamaño que se especifique, el cual debe ser estructurado en sectores, tal y como se explicará después.

La tarea se debe hacer en Java.

Descripción Detallada

- File System: Deben simular un File System. Esto implica que:
  - debe mantener una estructura de archivos
  - saber en que nivel de la estructura se encuentra uno en determinado momento. Debe ser claro la ruta actual, y esta debe estar disponible al usuario en todo momento.
  - debe permitir navegar entre los directorios del File System.
  - debe tener la seguridad correspondiente relacionada a no crear dos directorios o archivos con el mismo nombre en un directorio.
  - si se va a crear un archivo o directorio con el nombre de uno que ya existe debe preguntar si le desea “caer encima” al archivo/directorio ya existente.
  - debe avisar en el caso que el disco este lleno
- Funciones del File System: Los comandos o funciones a implementar son las siguientes:
  - CREATE: Este comando lo utilizaremos para crear un disco virtual. Los parámetros serán la cantidad de sectores y el tamaño del sector.
  - FILE: Crear un Archivo. Se le debe definir el contenido del archivo y el nombre y extensión de este.
  - MKDIR: Este comando crea un directorio en el directorio Actual. El parámetro es el nombre del directorio.
  - CambiarDIR: Permite cambiar el directorio actual. Me debe permitir irme a un directorio cualquiera de la estructura de directorios actual
  - ListarDIR: Lista los archivos y directorios dentro del directorio actual. Debe mostrar una diferencia clara entre los directorios y archivos.
  - ModFILE: Se puede seleccionar un archivo y cambiarle el contenido.
  - VerPropiedades: Permite ver las propiedades de un archivo. Nombre, Extensión, Fecha de Creación, Fecha de Modificación y tamaño.
  - ContFile: Para un determinado archivo se debe poder ver el contenido del archivo.
  - CoPY: Se implementarán 3 tipos de copies. (aplica a archivos o directorios)
    - un archivo con ruta “real” será copiado a una ruta “virtual” de MI File System.
    - un archivo con ruta “vitual” de MI File System será copiado a una ruta “real”
    - un archivo con ruta “vitual” de MI File System será copiado a otra ruta “virtual” de MI File System.
  - MoVer: Mover un archivo o directorio. Nótese que el MV sirve como rename, pues se puede mover al mismo directorio con otro nombre.
  - ReMove: Con este comando se eliminaran archivos. La eliminación puede ser normal para uno o varios archivos o recursiva en el caso de directorios.
  - FIND: Recibe un nombre de archivo o directorio y lista todas las rutas de MI File System con archivos de ese nombre. (Nótese que puede recibir de parámetro algo como “*.doc”)

    - TREE: Despliega, simulando un árbol, la estructura de archivos del File System. Debe estar visible siempre.
- La anterior es una explicación en términos generales de los comandos del sistema, sin embargo la forma es que lo implementen depende de la interfaz que hagan. Recuerden que lo importantes es que se puedan hacer las funciones descritas anteriormente
- DISCO VIRTUAL: Se debe crear un archivo del tamaño especificado en el create y manejar los sectores dentro de él. La forma de asignación de sectores a un archivo debe ser FIRST FIT. Además la asignación debe ser Enlazada. Sin embargo, el puntero no es necesario que lo tengan en el mismo sector. Lo pueden manejar en memoria.
- Cuando la aplicación termina se pierde el file System, sin embargo no deben eliminar el archivo que representa el disco.
- Dado el tipo de asignación puede haber fragmentación interna.
- Al ser enlazado, un archivo puede ser almacenado en disco aunque no estén todos lo sectores contiguos. La única razón por la que no se pueda crear un archivo es que no haya suficientes sectores como el archivo lo necesita. (Deben llevar un control de los sectores vacios)

Documentación

Se espera que sea un documento donde especifique:

- El análisis de resultados del programa. El Análisis debe resumir el resultado de la programación, qué sirve, qué no sirve y aspectos que consideren relevantes.
- Cómo compilar y ejecutar los programas.

Aspectos Administrativos

- El trabajo se debe de entregar la semana del 15 de Junio, por correo
- Deben entregar el código fuente junto con el ejecutable.
- Recuerde que oficialmente no se recibirán trabajos con entrega tardía.

SUERTE!!!