

## Proyecto # 2 (17 %)

Se desea que Ud. implemente un simulador de un Sistema Manejador de Disco Duro (SMD) usando el conjunto de instrucciones MIPS.

### 1 Sistema manejador de disco FAT

Los archivos normalmente se almacenan en disco, de manera que el manejo del espacio de disco es de interés fundamental para los diseñadores de sistemas. En todo disco siempre hay que diferenciar entre su estructura física y su estructura lógica.

La estructura física es inherente al disco y se crea cuando se construye el disco en la fábrica, asignándole un número determinado de caras, cilindros (pistas) y sectores.

La estructura lógica se crea cuando se formatea el disco y su función es organizar la superficie del disco para almacenar datos. La estructura lógica depende del sistema de operación que formatea el disco y, en algunos sistemas de operación, se divide el disco en un sector de arranque, una tabla de localización de archivos (**FAT**) y el área para los datos del usuario. El sector de arranque (boot sector) es el primer sector (512 bytes) y se encarga de comenzar el arranque del sistema de operación (o mostrar el mensaje de error “No es disco de sistema” cuando el disco no está preparado para arrancar un sistema de operación). Se reserva un espacio para la **FAT** y el resto se deja para almacenar los datos del usuario

#### 1.1 Clusters de un disco duro

El sector es la unidad mínima de información desde el punto de vista de la estructura física, pero el sistema de operación, no entiende de sectores y sólo trabaja con clusters.

Un cluster no es más que un conjunto de varios sectores contiguos del disco (y recuerde que un sector es un bloque de 512 bytes). El número de sectores

contiguos que tiene un cluster, es decir, el tamaño del cluster, varía en cada unidad de disco y depende del tamaño de la unidad (cuanto más grande sea la unidad, más grande será el tamaño del cluster), pero siempre es potencia de 2 (1, 2, 4, 8, etc). Como mínimo, cualquier archivo ocupa un cluster. En definitiva, el cluster es la unidad mínima de información desde el punto de vista de la estructura lógica del disco, que es lo que cuenta para el sistema de operación.

Los archivos se guardan en clusters, pero esos clusters no tienen por qué estar contiguos (consecutivos). Ahora bien, cómo sabe el sistema de operación cuál es la cadena de clusters de un archivo, por ejemplo, cuál es el primer cluster de un archivo y en qué cluster continúa?

La FAT (File Allocation Table; tabla de Asignación de archivos) es una tabla formada por elementos que se corresponden con cada uno de los clusters del disco. En definitiva, la FAT es una tabla que permite al sistema de operación utilizar clusters no consecutivos para almacenar los archivos. Si por cualquier motivo se corrompe la FAT posiblemente perderá gran parte de sus datos, ya que el sistema de operación no sabrá donde continua un archivo y donde termina. Es tal la importancia de la FAT, que normalmente se guardan 2 copias de la FAT para recuperar los datos en caso de que se corrompa una de las copias.

La unidad mínima de asignación de almacenamiento para un archivo es el cluster. Esto quiere decir que el espacio real ocupado por un archivo en disco será siempre múltiplo del tamaño del cluster. Además, cada cluster puede almacenar información de un solo archivo. Si no cabe en un solo cluster, se utilizarán varios. En DOS el tamaño del cluster es de 512 bytes.

Cuando un requerimiento de creación de archivos es recibido, el Sistema Manejador de Disco (SMD) busca un cluster libre, corta o divide el archivo en segmentos de tamaño de un cluster. Almacena estos segmentos en los clusters libres y realiza las actualizaciones necesarias.

Para mantener en el disco un sistema de archivos son necesarias dos estructuras de datos, el **Directorio** y la **FAT**.

## 1.2 FAT (File Allocate Table) y Directorio

La FAT lleva registro de todos los clusters que han sido asignados, así como los clusters libres. Su simulador ejecuta el comando y da respuesta si el comando lo amerita. Cada cluster tiene una única identificación (id) y son enumerados en secuencia. Por otro lado, el Directorio mantiene la asociación entre el nombre del archivo y el número de su primer cluster. Siempre que el sistema de operación pregunte por un archivo, el directorio es consultado para obtener el número de su primer cluster y luego se va a la FAT con este número y se encuentran el resto de los clusters asociados. Ver figura 1. En el ejemplo de la figura, el archivo “test” ocupa los clusters 217,618 y 339.

Para el manejo de los clusters podríamos usar, un valor 0(cero) en una entrada de la tabla FAT para significar cluster libre, un valor -1 como fin del archivo (EOF), un valor n no negativo y diferente de cero, representará el siguiente

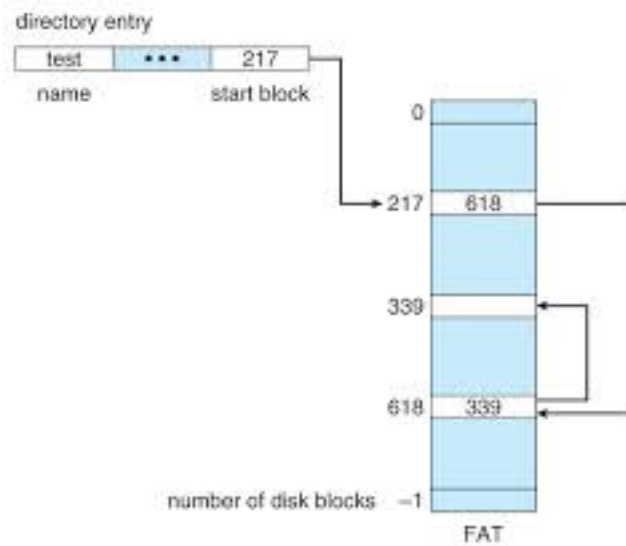


Figure 1: Directorio y FAT

cluster que forma parte del archivo.

## 2 Especificación del Proyecto

1. El espacio del disco a simular es un bloque de memoria de tamaño 1Kbytes.
2. Cada cluster es de tamaño 4bytes. Así que el disco tiene 256 clusters.
3. El tamaño de la FAT y Directorio le es dejado a Ud. Asegúrese de que pueda acomodar suficientes archivos en el disco. Estas tablas NO necesitan estar en el espacio del disco. Los puede colocar en otro bloque de memoria.
4. Los detalles de las estructuras, como por ejemplo el número de bytes por entrada, le es dejado a Ud.
5. En su simulación los archivos serán representados como strings.

## 3 Como se debe ejecutar el simulador

1. Su simulador da por consola un prompt como por ejemplo ">" o cualquier otro.
2. El usuario entra uno de los comandos listados abajo.
3. Su simulador ejecuta el comando y da respuesta si el comando lo amerita.

4. Imprime el signo de prompt de nuevo.
5. Se ejecuta hasta que el usuario no desee seguir ejecutando el programa.

## 4 Lista de funciones

1. **crear “nombrearchivo”**: crea el archivo de nombre **nombrearchivo** en el Sistema Virtual Manejador de Disco (SMD) .Para ello, actualiza las tablas y guarda el contenido del archivo en los clusters necesarios de memoria. La información a guardar en los clusters será leída de un archivo con nombre **nombrearchivo**.  
El algoritmo para la asignación de clusters libre puede ser el primer cluster libre que se encuentre.
2. **imprimir “nombrearchivo”**: imprime el contenido del archivo por consola.
3. **copiar “archivofuente” “archivodestino”**: copia el contenido del primer archivo en el segundo archivo.
4. **ren “nombre1” “nombre2”**: renombra un archivo. Al archivo de nombre **nombre1** le asigna el nombre **nombre2**.
5. **sizeof “nombrearchivo”**: tamaño del archivo. Devuelve el tamaño del archivo en unidades de clusters y en unidades de bytes.
6. **buscar “string”**: busca en el directorio el primer archivo que contenga el string solicitado. Devuelve el nombre del archivo que lo contiene.
7. **dir**: lista los archivos del disco y su tamaño en unidades de bytes.

### Observaciones

1. No se permite los comandos de crear y copiar cuando el espacio en disco está lleno. Se debe escribir por consola un mensaje de error.
2. El comando crear debe dar un error si el archivo ya existe.
3. Los comandos de eliminar, imprimir, copiar y renombrar archivos con archivos que no existen causará un mensaje de error.

**Fecha de entrega:** Viernes 30-05-2014. Debe, en horas de clase, entregar el código impreso, y el día anterior hacer submit del archivo a Moodle.