Sistemas Operativos I 2023
Recuperatorio del Segundo parcial

Temario

- Lenguaje C. Sistema Operativo Xinu. Linux.
- Comunicación inter-procesos. Drivers de dispositivos de E/S. Memoria virtual. Manipulación de archivos utilizando operaciones del sistema de archivos.

Software y Hardware

El parcial se realiza sobre las computadoras de laboratorio. Utilizar la versión de XINU en donde desarrolló el driver del teclado y lo utiliza en el juego gálaga. También puede utilizar como referencia los programas de los TPs relativos a comunicación entre procesos, memoria virtual y sistemas de archivos.

Ejercicio 1.

a.

Implemente en Linux dos programas:

1. Un programa crea memoria compartida, y carga en la memoria compartida la foto cat.pgm, usando funciones para acceder a archivos en el sistema de archivos.
2. Un segundo programa lee de la memoria compartida la foto, y la guarda en un nuevo archivo de manera invertida.

- Para invertir la imagen se debe mantener la cabecera intacta, y luego deberán estar los bytes de los píxeles invertidos (el último byte de pixel de cat.pgm será el primer byte de pixel en cat2.pgm, etc).

El formato pgm de la imagen de ejemplo es sencillo:

- Los primeros 15 bytes son la cabecera del formato del archivo de imagen.
- Los siguientes bytes son los valores de cada pixel de la imagen.

Utilice memoria compartida entre los dos programas.
El tamaño de la memoria compartida es estático y conocido: 50261 bytes (que es el tamaño del archivo cat.pgm)

Las funciones de C para crear un archivo nuevo y escribir 5 bytes son (suponga que imagen es un ARREGLO).

```
#include <unistd.h>

const char *nuevo = "cat2.pgm";
fd = open(nuevo, O_RDWR | O_CREAT | O_TRUNC, 0666); /* le pide al sistema operativo crear un archivo nuevo */
write(fd, &imagen[6], 5);    /* escribe en el archivo con descriptor fd 5 bytes, a partir del septimo byte del arreglo imagen[ ] */
```

b. Agregue al SEGUNDO programa sentencias para que emita por pantalla la dirección virtual del inicio del segmento de memoria compartida. ¿Cuál es esa dirección? (recuerde que son direcciones de 64 bits).

c. 1. Utilizando /proc/PID/maps del segundo programa en ejecución indique las direcciones virtuales del segmento donde se encuentra la memoria compartida (como tip podría observar maps antes de obtener acceso a la memoria compartida, y luego de haber tenido acceso, y detectar cuál es o son los segmentos nuevos).

2. Indique tambien a qué segmento de memoria del proceso pertenece.

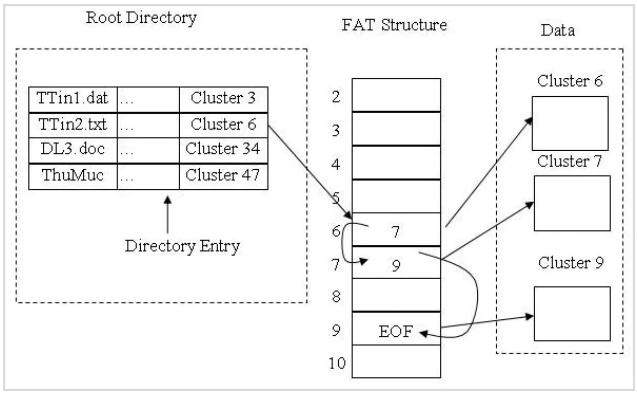
Ejercicio 2.

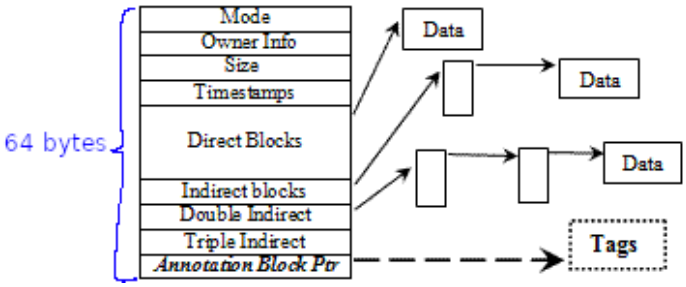
Modifique su entrega de XINU que implementa el driver del teclado. Agregue un programa al shell llamado "teclado", que verifique el uso del teclado y system call read(). El programa "teclado" debe borrar toda la pantalla gráfica, solicitar 20 pulsaciones de tecla al sistema operativo utilizando read, y mostrar los códigos hexadecimales de las 20 pulsaciones de teclas devueltas por el sistema operativo. La función que limpia la pantalla se llama paint_screen();

Ejercicio 3.

Responda:

1. ¿Qué problemas podría ocasionar a su sistema XINU que un programa no utilice apropiadamente el paradigma open(), read(), close()?. Por ejemplo, si no se utiliza apropiadamente en la resolución del ejercicio 2. Justifique su respuesta.
2. Es de conocimiento general que los sistemas Windows de empresas y personas sufren ataques de distintas índoles todo el tiempo. Millones de agujeros de seguridad en Windows se han reportado en internet en los últimos 30 años. Si usted desarrolla dos programas en Windows (prog1 y prog2), y dos programas en XINU (prog1 y prog2), responda: ¿en qué sistema es más sencillo realizar un ataque desde prog1 a prog2?. Justifique su respuesta.
3. Sean las siguientes estructuras de dos sistemas de archivos: FAT y un FS UNIX.





i-nodo

- Suponga que el i-nodo de un archivo deseado ya está en RAM en el UNIX FS.
- Suponga que la FAT está en RAM.
- Suponga que en ambos sistemas el tamaño máximo para un archivo son iguales, y los bloques de datos son del mismo tamaño.

Si sólo se desea conocer los números de bloques en el disco que corresponden a los datos del archivo deseado (no su contenido), ¿en cuál de los dos sistemas de archivos sería más veloz obtener todos los números de bloques de datos que pertenecen al archivo?. Justifique.