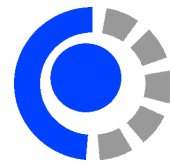




Sistemas Operativos I 2023

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 memoria virtual y sistemas de archivos.

Ejercicio 1.

a.

Implemente en Linux un programa que invierta la foto cat.pgm que se adjunta al parcial, y genere un NUEVO archivo llamado cat2.pgm con la imagen invertida.

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.

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

Utilice memoria dinámica: cargue la imagen original en un segmento de memoria solicitado al sistema operativo. El programa debería funcionar para cualquier tamaño de archivo, no sólo el utilizado aquí en el parcial.

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 programa sentencias para que emita por pantalla la dirección virtual del inicio del segmento de memoria dinámico solicitado al sistema operativo. ¿Cuál es esa dirección? (recuerde que son direcciones de 64 bits).
- c. 1. Utilizando `/proc/PID/maps` indique las direcciones virtuales del segmento donde se encuentra la memoria dinámica solicitada al sistema operativo (como tip podría observar maps antes de solicitar la memoria al SO, y luego de haberla solicitado, y detectar cuál es o son los segmentos nuevos).
2. Indique también a qué segmento de memoria del proceso pertenece.

Ejercicio 2.

Modifique su programa en XINU que implementa el driver del teclado, para que la sección upper-half del driver se comunice con la sección lower-half utilizando algún mecanismo de comunicación entre procesos. Es decir, en vez de utilizar el buffer y la sincronización implementada en el trabajo práctico, reemplace lo anterior con algún mecanismo de comunicación entre procesos provisto por XINU.

Ejercicio 3.

Responda:

- Si las páginas del sistema de memoria virtual en los sistemas Linux de los laboratorios son de 4KB, ¿cuáles son los 12 bits menos significativos de la dirección física que corresponde a la dirección virtual obtenida en el ejercicio 1.b. ? Es decir, luego de que el sistema realizó la traducción de la dirección virtual a física. Utilice el sistema hexadecimal para especificar la respuesta.
- En C en Linux existen las funciones `shm_open()`, `ftruncate()` y `mmap()` para crear un segmento de memoria compartida que luego se puede compartir con otros procesos para comunicación entre procesos. Indique en XINU cuáles serían las funciones en C equivalentes, y si no existen, explique cómo se logra ese mecanismo de comunicación entre procesos.
- Sean los siguientes diagramas de un UNIX file system:

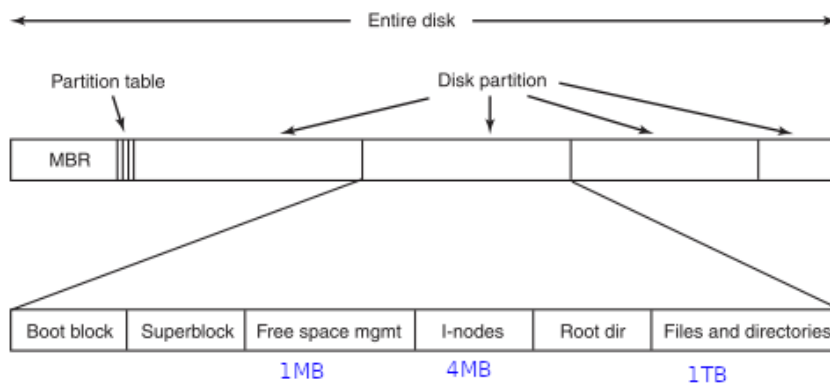
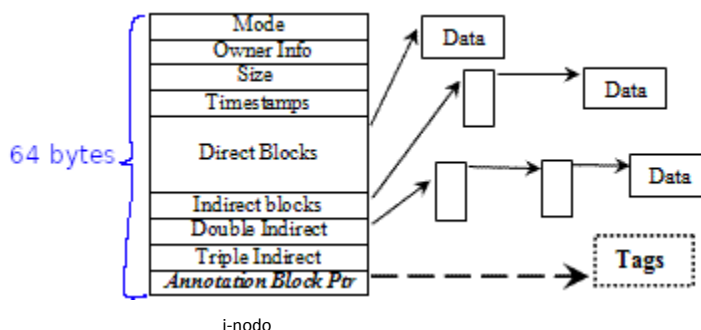


Figure 4-9. A possible file-system layout.



¿Cuántos archivos máximo puede almacenar este sistema de archivos?