# Entrega 6 - Sistemas Operativos II

Delfina Martín - Ignacio Litmanovich - Agustín Diaz

Abril 2021

# Ejercicio 1

## Apartado a

Verdadero. **raise** manda una señal al hilo actual (o llamante) y **kill** manda una señal a un proceso o hilo a través del pid que se pasa como argumento (no necesariamente diferente al hilo actual). Para una señal *sig* dada, el raise definido en términos de kill queda

$$\text{raise}(sig) = \text{kill}(\text{getpid}(),\ sig)$$

donde getpid() devuelve el pid del hilo actual.

## Apartado b

Falso. Como raise solamente puede enviar una señal al hilo actual (o llamante), no es posible definir kill en términos de este: no hay cómo indicar el hilo/proceso al cual se quiere señalizar (si es que este no coincide con el actual).

## Apartado c

Falso. Signal se encarga de asociar señales a manejadores (que determinan qué hacer cuando se intercepta la señal). Existe un handler por defecto SIG_IGN (signal ignore) con el cual se ignora la señal interceptada. Este es un ejemplo de una invocación a signal en la cual el handler no envía señales.

## Apartado d

Falso. La senal SIGSEGV *(segmentation violation)* sí indica una violación de segmento o referencia inválida a memoria virtual, pero un proceso podría enviarla sin que necesariamente haya una violación de segmento real. [Adjuntamos pequeño programa ejemplo]

# Ejercicio 2

Cuando el manejador de señales retorna, la ejecución continua desde el punto en donde se lanzó la señal, y se re ejecuta la última instrucción ejecutada. Lo cual vuelve a invocar al manejador, reinicia el ciclo y se convierte en un bucle infinito.

El manejador por defecto de SIGSEGV por su parte, se encarga de volcar la memoria (dump core) y de terminar la ejecución del programa. [Adjuntamos pequeño programa ejemplo]

# Ejercicio 3

El mecanismo de intercambio presentado en la sección 31.1.3 consiste en suspender un proceso y almacenar una copia de su imagen en memoria en almacenamiento secundario para luego restaurarlo. Si el proceso en cuestión tiene operaciones de entrada/salida pendientes, los resultados de estas operaciones se almacenan en buffers (almacenamiento temporal) que se encuentran en el espacio de sistema operativo.

El mecanismo presentado en la sección 32.2 en contraposición, aprovecha el hecho de que ciertas porciones de código no se ejecutan secuencialmente y separa su texto (e incluso sus datos) en diferentes segmentos. Los segmentos que no se usan en largos períodos de tiempo son enviados al área de intercambio (swap).

El primer mecanismo resulta costoso frente al segundo debido al tiempo que toma guardar una copia completa de la imagen de un proceso en almacenamiento secundario. Además, la carga administrativa es considerable teniendo en cuenta la logística del almacenamiento temporal que se utiliza para no generar inconsistencias en las operaciones de E/S. El segundo mecanismo presenta la ventaja de que los segmentos se pueden mandar al swap según la política LRU (least recently used), es decir, se envían los segmentos que menos probabilidad tienen de usarse. Más aún, los segmentos de texto (con acceso de sólo lectura) se pueden copiar al disco una única vez y marcar como no presentes en las tablas de segmentos. Y esto puede mejorar enormemente su desempeño.

Una última ventaja del mecanismo de intercambio parcial tiene que ver con las bibliotecas linkeadas dinámicamente: el SO no necesita cargar los archivos en memoria si la biblioteca reside en disco como una imagen directa de su representación en memoria, basta con mantener una referencia a ella.

# Ejercicio 4

Suponemos que todas las páginas corresponden a un mismo proceso por simplicidad.

| Pagina | 1 | 2 | 3 | 4 | 1 | 2 | 5 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Marcos | 1 | 1 | 1 | 1 | 1 | 1 | 5 | 5 | 5 | 5 | 4 | 4 |
| | | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 5 |
| | | | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 |
| | | | | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 |
| M/H | M | M | M | M | H | H | M | M | M | M | M | M |

Tabla 1: Política FIFO en un sistema de 4 páginas físicas

**Resumen del desempeño**: 10 Misses (Fallos de página) y 2 Hits (Aciertos).