Sistemas Operativos I. Primer examen parcial. 2025. Facultad de Informática. Universidad Nacional del Comahue

NOTA: para la resolución del examen se puede utilizar como material de referencia sus archivos de solución de los trabajos prácticos en sus cuentas Linux, o el material disponible en la web de la materia. http://se.fi.uncoma.edu.ar/so

En ningún caso se pueden utilizar herramientas online (google, chat, gpt, deepseek, etc).

Ejercicio 1.
Desarrollar un programa en Linux que indique si una palabra o número ingresado por el usuario es palíndromo. Una palabra o frase es palíndromo si se lee igual en un sentido que en el otro (por ejemplo; ana, anna, otto, neuquen son palíndromos). Si se trata de números en lugar de letras, se llaman capicúas (121, y 3333 son capicúas).

Específicamente, el programa debe ir solicitando pulsaciones de teclado al usuario, y colocando la entrada del usuario en un arreglo. Hasta que el usuario presione enter (el ENTER tiene código 13 en base 10).

Luego, se debe analizar el arreglo para ver si la entrada del usuario es palíndromo.
Al finalizar el análisis el programa debe mostrar la entrada del usuario y una leyenda que indique si es palíndromo o no.

Ejercicio 2.
Portar el programa del Ejercicio 1. a XINU, con la siguiente modificación: el programa principal debe crear dos procesos: procA y procB. El proceso procA pide la entrada al usuario. El segundo proceso procB computa el posible palíndromo y muestra el resultado. El programa principal deberá esperar a que procA finalice antes de poner a ejecutar a procB. Integre el programa al shell de XINU y verifique su funcionamiento.

NOTA1: el system call **receive()** sin argumentos puede ser utilizado en XNU para esperar a que un proceso hijo finalice (receive() bloquea al proceso hasta que un proceso hijo finalice).
NOTA2: el arreglo donde quedará la entrada del usuario debe ser global para que todos los procesos puedan accederlo (lo mismo para otras variables compartidas).

IMPORTANTE: Para la resolución utilice UNICAMENTE el siguiente codigo fuente de XINU:
https://se.fi.uncoma.edu.ar/so/misc/xinu-pc.tar.gz
(SI REUTILIZA el código fuente de XINU que usó durante los trabajos prácticos SE ANULA LA ENTREGA).

Se entrega todo el código fuente de XINU, no sólo el fuente .c . Antes de crear un zip o tar.gz de la carpeta xinu-pc realice un make clean en compile/

(ejercicio 3 en siguiente hoja)

Sistemas Operativos I. Primer examen parcial. 2025. Facultad de Informática. Universidad Nacional del Comahue

Ejercicio 3. Responda en un archivo .txt

a. Verdadero o Falso (justificar en ambos casos, en verdadero o falso):

1. Zombie es un estado posible del sistema operativo Android.
2. Tanto Windows como Linux utilizan el algoritmo de planificación de CPU CFS, porque son sistemas operativos más complejos.
3. En el algoritmo de FCFS podría suceder starvation (inanición).

b. COMPLETAR

En XINU :

El estado de un proceso SLEEPING corresponde al estado ……..…….. en “la teoría”

El estado de un proceso SUSPENDED corresponde al estado ……. En “la teoría”

El estado de un proceso CURRENT corresponde al estado …… en la “teoría”.

c. Estado de procesos en XINU.

Suponga que en una computadora con una única CPU y XINU existe un proceso que tiene un bucle ‘for’ infinito, que tiene más prioridad que los demás, y que nunca pasa a un estado de durmiendo o bloqueado….

(Indique cuales opciones debajo son correctas -justifique las correctas-)

1. Los demás procesos eventualmente obtendrán la CPU porque el sistema es también round robin.
2. Los demás procesos en estado de listo nunca obtendrán la CPU y se producirá startvation.
3. Los demás procesos podrían intentar hacer un kill() del proceso con alta prioridad para matarlo.
4. Algún proceso obtendrá la CPU luego de 2ms, porque el QUANTUM en XINU es 2ms.