Informe Mini TP3

Memoria y Planificadores

López Ciro Martín

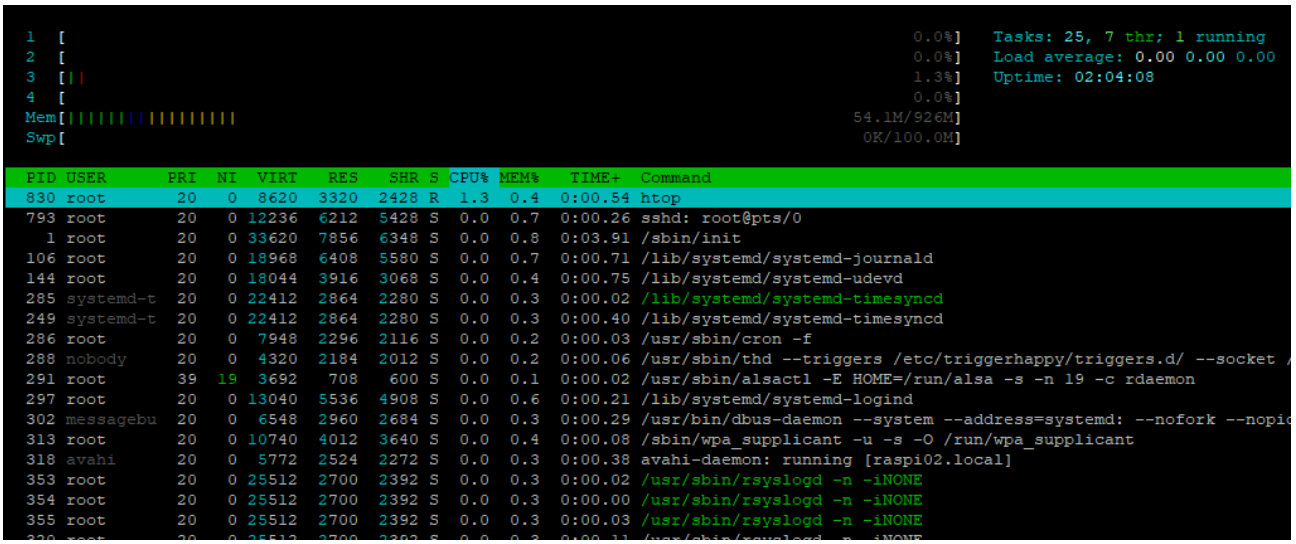
Sistemas Operativos y Redes 1

2do cuatrimestre 2025

Profesores: Andres Rojas, Leandro Dikenstein

Enunciado del Mini TP

Hace poco la materia SOR se compró un Raspberry pi, el cual tiene un procesador ARM v8, 1.2 Ghz, 4 núcleos y 1 Gb de Ram. Para instalar el sistema operativo le compró una tarjeta micro sd, 32 gb clase 10 marca Lexar. Instaló Ubuntu mate, pero no quedó muy conforme. Decide consultarle a sus amigos de SOR2 y ellos le aconsejan instalar Raspbian Buster Lite (de 32 bits).
Una vez instalado el S.O. comprueba el estado de los procesos con htop y ve lo siguiente:



Aparentemente el sistema operativo consume 54.1 MB, averigua también que Raspbian hace paginación de a 8KB. Decide compilar un programa en C con funciones recursivas y nota que cuando ejecuta en la consola
root@raspi02:~# ./ejecutable_A este consume 128.05 MB.

Memoria
Responder:
Analizando los datos puedo saber que:
ejeecutable_A: 128.05 MB= 128.05 × 2^20 bytes = 128.05×220 bytes = 134270156.8 bytes.
Paginación:8 KB = 8 × 1024 bytes = 8192 bytes

a) ¿Cuántos bits necesito para dar un número de página del proceso ejecutable_A (el de las funciones recursivas)?
Necesitas suficientes bits para indexar todas las páginas asignadas:
bits página = [log2(número de páginas)].
Número de páginas = 16391
No nos sirve el log mas cercano si es que es menor al valor que buscamos, por eso no usamos 14 bits y sí 15 bits.
2^{14}=16384 (insuficiente), 2^{15}=32768 (suficiente).

Por tanto se necesitan **15 bits.**
Respuesta: 15 bits para el número de página del proceso.

b) ¿Cuántos bits necesito para el desplazamiento dentro de un frame?
Sabiendo que el tamaño de página es de 8192 bytes, entonces tendríamos que hacer log2(8192), dándonos como respuesta la cantidad de bits: **13 bits, para el desplazamiento dentro de un frame.**

c) ¿Cuántas páginas se asignan al proceso ejecutable_A?
Si debido a los bytes usados por el proceso por los bytes por página puedo saber cuántas páginas se le asignaron al proceso.
Bytes del proceso = 128.05×2^20=134270156.8 bytes.
Bytes por página = 8192
Cálculo de páginas: 134270156.8 / 8192=16390.4
Redondeamos hacia arriba dándonos un total de 16391 páginas para el proceso ejecutable_A.

d) ¿Hay fragmentación? Justificar.
Sí, existe fragmentación interna: la última página asignada no se utiliza completamente (porque el proceso no tiene un múltiplo exacto de tamaño de página).

Calculamos el sobrante concreto:
Bytes asignados = 16391 × 8192=134275072 bytes.
Bytes realmente usados = 134270156.8 bytes.
Diferencia = 134275072−134270156.8=4915.2 bytes.

Tenemos en total 4915.2 bytes de desperdicio en la última página.

Planificador

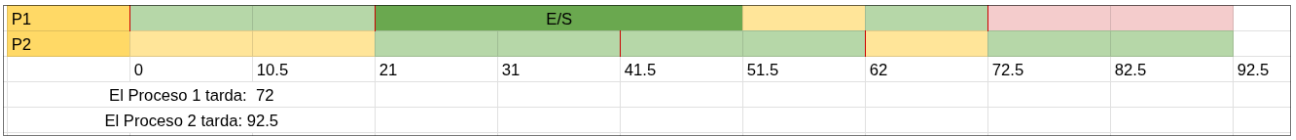
Supongamos que el sistema ejecuta 2 procesos de las siguientes características:
PROCESO 1 : Ejecuta 20 unidades de tiempo, luego efectúa una E/S sobre disco, luego ejecuta 10 unidades de tiempo. y termina.
PROCESO 2 : Ejecuta 60 unidades de tiempo y termina.
El método de selección de la cola de listos es el Round Robin asignándole a cada proceso un quantum de 20 unidades de tiempo. Una operación de E/S sobre disco tarda el equivalente 30 unidades de tiempo.
Suponer que ambos procesos llegan al mismo tiempo a la cola de listos y que el planificador decide ejecutarlos y planificarlos sobre el mismo núcleo.

Observación**:** tiempo de conmutación de tarea (context switch) de 0.5 unidades de tiempo

- Representar la planificación en un diagrama de Gantt e informar el tiempo de retorno de ambos procesos por separado. Puede realizar la planificación en una

planilla de cálculo. Justificar su planificación con el seguimiento paso por paso de Round Robin.

P1: Proceso 1.
P2: Proceso 2.



Justificación:

Paso 1: Inicialización (t = 0)
Llegan los dos procesos a la cola.
El planificador selecciona P1 para ejecutar.

Context Switch de 0.5 unidades: se carga P1.
Tiempo acumulado: 0.5

Paso 2: Ejecución de P1,
P1 ejecuta 20 unidades de tiempo.
Tiempo acumulado: 0.5 + 20 = 20.5

Al finalizar, P1 solicita una operación de E/S.

P1 se bloquea.

Context Switch de 0.5 unidades para cambiar a P2.
Tiempo acumulado: 20.5 + 0.5 = 21.0

Paso 3: Primera ejecución de P2
P2 ejecuta 20 unidades de tiempo.
Tiempo acumulado: 21.0 + 20 = 41.0
P2 vuelve al final de la cola de listos.

Context Switch de 0.5 unidades.
Tiempo acumulado: 41.0 + 0.5 = 41.5

Paso 4: Segunda ejecución de P2
P1 está bloqueado por lo que se ejecuta P2 que es el único proceso en la cola de listos.

P2 ejecuta 20 unidades de tiempo.
Tiempo acumulado: 41.5 + 20 = 61.5

P2 vuelve al final de la cola de listos.

Context Switch de 0.5 unidades.
Tiempo acumulado: 61.5 + 0.5 = 62

Paso 5: Vuelve P1 ya terminó de usar E/S y está en la cola de listos. Tanto P1 como P2 están en la cola de listo.s

Round Robin selecciona P1 (primero en la cola).

P1 ejecuta sus 10 unidades restantes.
Tiempo acumulado: 62+ 10 = 72

P1 termina completamente en t = 72

Context Switch de 0.5 unidades para cambiar a P2.
Tiempo acumulado: 72 + 0.5 = 72.5

Paso 6: Ejecución final de P2
P2 ejecuta sus 20 unidades restantes.
Tiempo acumulado: 72.5 + 20 = 92.5

P2 termina completamente en t = 92.5.

Representación de la cola:
t = 0.0
[P1, P2] ← Ambos procesos llegan juntos

t = 0.5 (después de context switch)
[P2] ← P1 sale para ejecutar, P2 queda en cola

t = 20.5 (P1 termina quantum, solicita E/S)
[P2] ← P1 se bloquea (va a E/S), P2 sigue en cola

t = 21 (después de context switch)
[] ← P2 sale para ejecutar, cola vacía

t = 41 (P2 termina quantum)
[P2] ← P2 vuelve al final de la cola

t = 41.5 (después de context switch)
[] ← P2 sale para ejecutar nuevamente

t = 50.5 (P1 termina E/S)
[P1] ← P1 termina E/S y entra a cola de listos

t = 61.5 (P2 termina quantum)
[P2, P1] ← P2 vuelve al final, P1 ya estaba en cola

t = 62.0 (después de context switch)
[P2] ← P1 sale para ejecutar, P2 queda en cola

t = 72 (P1 termina ejecución)
[P2] ← P1 termina y desaparece, P2 sigue en cola

t = 72.5 (después de context switch)
[] ← P2 sale para ejecutar, cola vacía

t = 92.5 (P2 termina ejecución)
[] ← Sistema vacío, todos los procesos terminaron

Dificultades encontradas:
Tuve que repasar cómo funcionaba el método de selección Round Robin, sobre todo la parte en la que el proceso 1 se bloquea para efectuar una E/S en el disco. Primeramente lo habia tomado como un proceso más del CPU pero luego me di cuenta del error: