

Resolucion ejercicio 36

36. Dado un esquema de paginación, donde cada dirección hace referencia a 1 byte, se tiene un tamaño de dirección de 32 bits y un tamaño de página de 2048 bytes. Suponga además un proceso P1 que necesita 51358 bytes para su código y 68131 bytes para sus datos.

Tamaño de dirección: **32 bits** → entonces el tamaño del espacio de direcciones es:

- 2^{32} bytes = 4294967296 bytes = **4GiB**

Cada dirección apunta a **1 byte**

Tamaño de página: **2048 bytes (2 KiB)**

Proceso P1:

- Código: **51358 bytes**
- Datos: **68131 bytes**

- a. De los 32 bits de las direcciones ¿Cuántos se utilizan para identificar página? ¿Cuántos para desplazamiento?
- Como sabemos que la pagina tiene un tamaño de 2048 bytes (2^{11} bytes) y que cada direccion tiene 32bits
 - **11 bits para el desplazamiento**
 - **31 - 11 = 21bits para el numero de pagina**
- b. ¿Cuál sería el tamaño lógico máximo de un proceso?
- Se refiere al tamaño máximo que *podría* tener un proceso dado el tamaño de dirección lógico del sistema
 - En paginación, el tamaño lógico máximo de un proceso está determinado por cuántas direcciones virtuales diferentes puede generar el proceso

$$2^{32}bytes = 4294967296bytes = 4GiB.$$

c. ¿Cuántas páginas máximo puede tener el proceso P1?

- Como tengo 21 bits para identificar las paginas, como maximo voy a poder representar $2^{21} = 2097152$ paginas.

d. Si se contaran con 4GiB (Gibibytes) de RAM, ¿Cuántos marcos habría disponibles para utilizar?

- La RAM tiene una capacidad de 4GiB = 2^{32} bytes
- Sabemos que el tamaño del marco = tamaño de la pagina = 2048 = 2^{11}
- Entonces puedo tener una cantidad de marcos de:

$$\frac{2^{32}}{2^{11}} = 2^{21} = 2097152$$

- Dio lo mismo que la cantidad de paginas → solo porque la RAM es del mismo tamaño que el espacio de direcciones

e. ¿Cuántas páginas necesita P1 para almacenar su código?

- Código ocupa: 51358bytes
- Tamaño de pagina: 2048 bytes

$$\lceil \frac{51358}{2048} \rceil = 26$$

f. ¿Cuántas páginas necesita P1 para almacenar sus datos?

- Datos ocupa: 68131bytes
- Tamaño de pagina: 2048 bytes

$$\lceil \frac{68131}{2048} \rceil = 34$$

g. ¿Cuántos bytes habría de fragmentación interna entre lo necesario para almacenar su código y sus datos? Ayuda: Recuerde que en una misma página no pueden convivir código y datos.

- Para el código:
 - 26 páginas × 2048 = 53248 bytes asignados

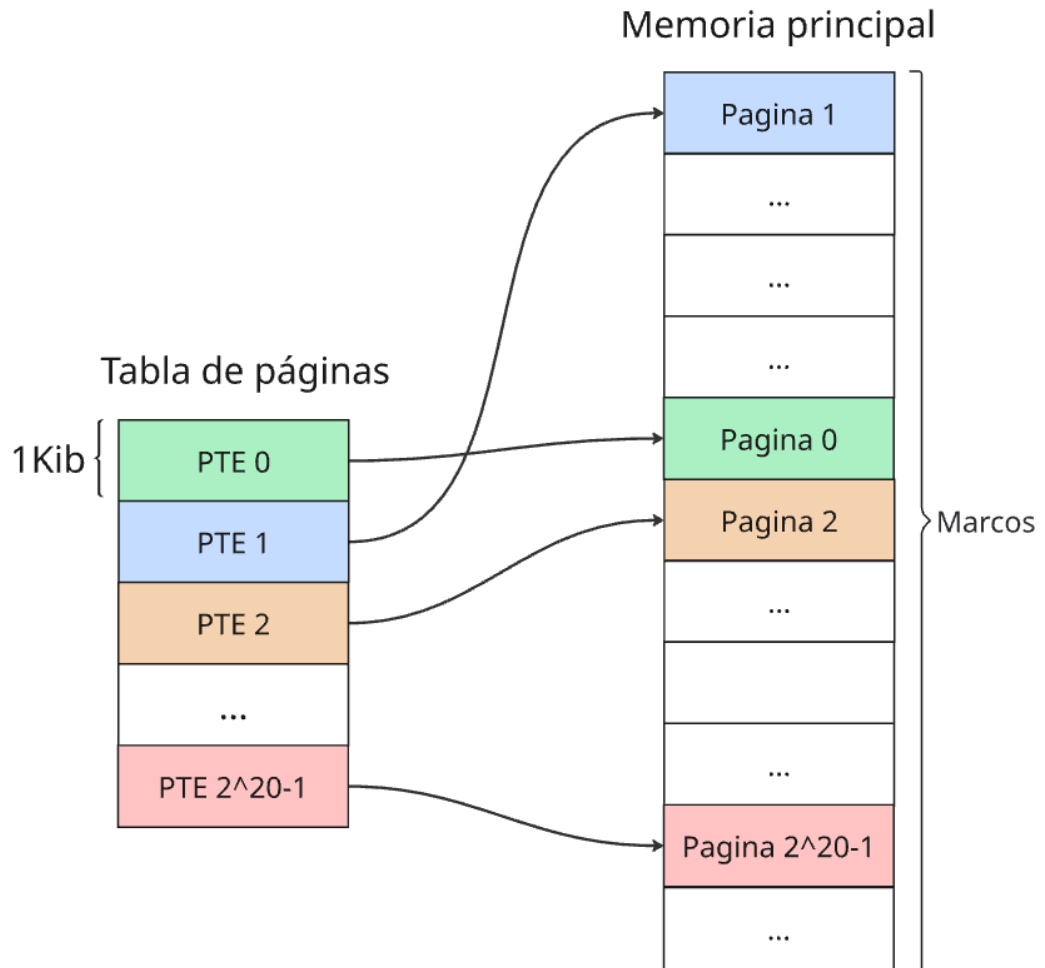
- $53248 - 51358 = 1890$ bytes desperdiciados
- Para los datos:
 - $34 \text{ páginas} \times 2048 = 69632$ bytes asignados
 - $69632 - 68131 = 1501$ bytes desperdiciados
- Total: $1890 + 1501 = 3391$ bytes de fragmentación interna

h. Asumiendo que el hardware utiliza tabla de páginas de un (1) nivel, que la dirección se interpreta como 20 bits para identificar página y 12 bits para desplazamiento, y que cada Entrada de Tabla de Páginas (PTE - Page Table Entry) requiere de 1 KiB. ¿Cuál será el tamaño mínimo que la tabla de páginas del proceso P1 ocupará en RAM?

- Dirección: 20bits para pagina + 12bits desplazamiento
 - Voy a tener $2^{20} = 1048576$ cantidad de paginas posibles
 - Cada pagina tiene un tamaño de $2^{12} = 4096$ bytes
- Cada entrada ocupa $1\text{KiB} = 1024\text{bytes} = 2^{10}$
 - **COMO ES UNA TABLA DE PAGINAS DE 1 NIVEL** tenemos 2^{20} paginas, entonces sabemos que vamos a tener 2^{20} entradas
 - Entonces podemos calcular

$$2^{20} \times 2^{10} = 2^{30} = 1GiB$$

- La tabla de paginas del P1 ocupa **1GiB**



i. Asumiendo que el hardware utiliza tabla de páginas de dos (2) niveles, que la dirección se interpreta como 10 bits para identificar primer nivel, 10 bits para identificar el segundo nivel, 12 bits para desplazamiento y que cada Entrada de Tabla de Páginas (PTE) requiere de 1 KiB. ¿Cuál será el tamaño mínimo que la tabla de páginas del proceso P1 ocupará en RAM?

- Informacion:
 - 10 bits primer nivel $\rightarrow 2^{10} = 1024$ entradas
 - 10 bits segundo nivel $\rightarrow 2^{10} = 1024$ entradas
 - 12bits para el desplazamiento \rightarrow la pagina va a tener un tamaño de $2^{12} = 4096$ bytes
 - Cada entrada ocupa 1KiB = 1024bytes

- Cálculos:

Cantidad de paginas que necesita P1 para código: $\lceil \frac{51358}{4096} \rceil = 13$

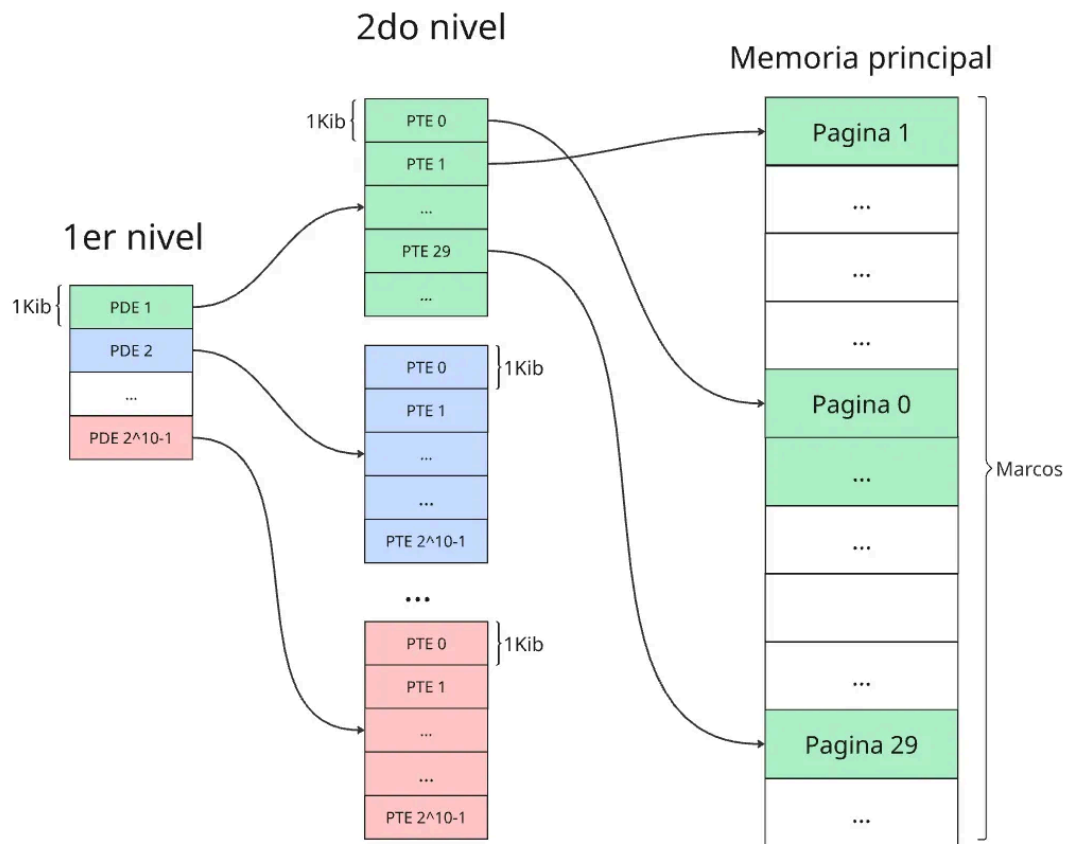
Cantidad de paginas que necesita P1 para datos: $\lceil \frac{68131}{4096} \rceil = 17$

Cantidad total de paginas = $17 + 13 = 30$ paginas

Primer nivel de la tabla tiene 1024 entradas y como cada entrada requiere 1KiB $\rightarrow 1024 \times 1KiB = 1MiB$

Cada tabla del segundo nivel tiene 1024 entradas y como cada entrada requiere 1KiB $\rightarrow 1024 \times 1KiB = 1MiB$

- Entonces como tenemos que almacenar 30 paginas con solo **una tabla del segundo nivel** nos va a alcanzar para almacenar las paginas del proceso 1. Por lo tanto como mínimo va ocupar 1MiB (tabla del primer nivel) + 1MiB (tabla del segundo nivel) = **2MiB**



- j. Dado los resultados obtenidos en los dos incisos anteriores (h, i) ¿Qué conclusiones se pueden obtener del uso de los 2 modelos de tablas de

página?

- Tabla de un nivel → ineficiente en cuanto espacio
- Tabla de dos niveles → mas compleja de mantener, pero ahorra espacio. Además requiere mas accesos a memoria para poder resolver una direccion:



Tips para el ejercicio y para el parcial:

- Trabajar siempre con potencias de 2.
- Siempre hacer las cuentas en las mismas unidades (bits, bytes, KiB, MiB, etc.)
- Si el tamaño no es múltiplo de la página, siempre se necesita una página completa extra (redondeamos para arriba)
- **Traer calculadora!!!!**