

Ejercicios

1. Considere un sistema con un esquema de manejo de memoria de paginación de dos niveles con un espacio de direcciones virtuales de 128MB, direcciones físicas de 32 bits y páginas de 4KB. Cada entrada en la tabla de páginas ocupa 32 bits.

¿Cuál es el tamaño máximo de un proceso ?

¿Cuál es el número de bits necesarios para direccionar la tabla de páginas del primer nivel?

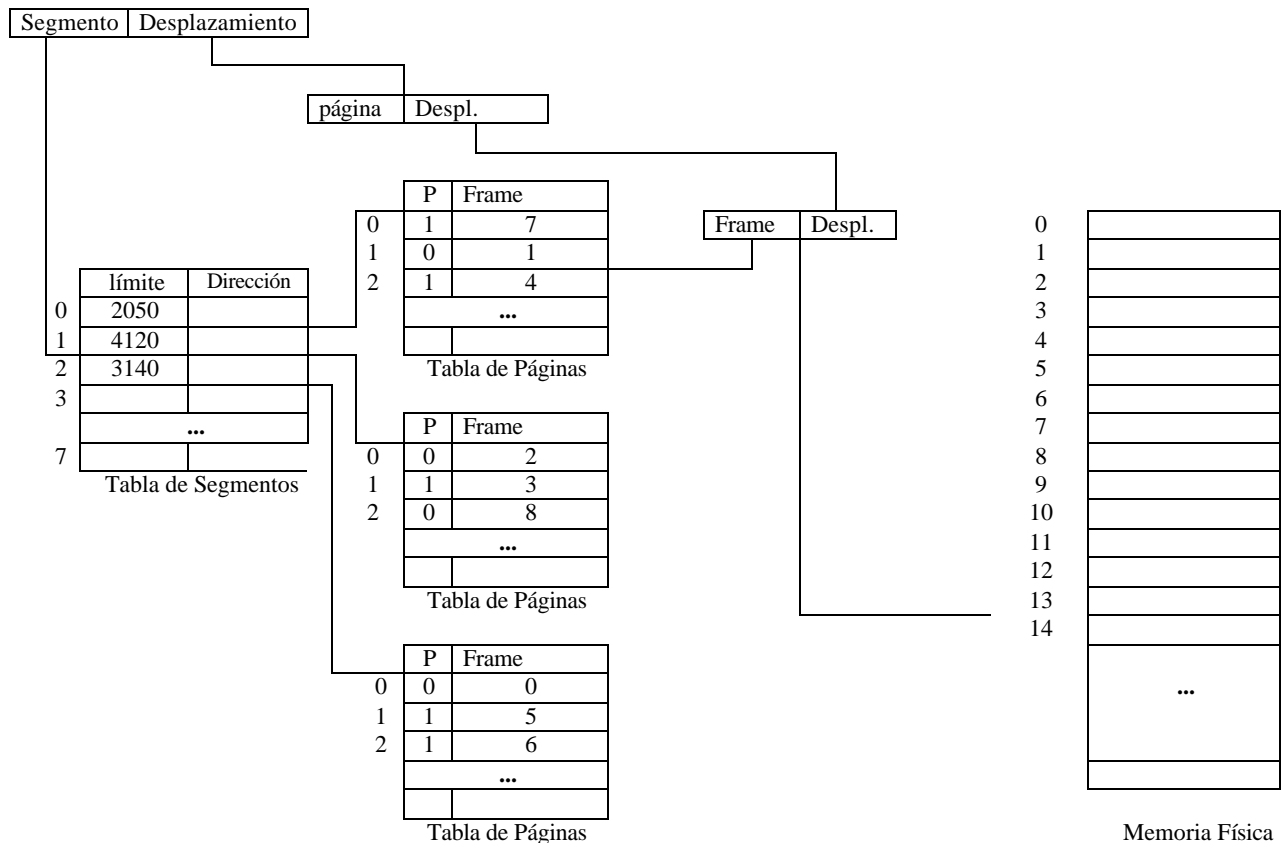
¿Cuál es el número máximo de entradas en la tabla de páginas del segundo nivel?

¿Cuál es el número de bits necesarios para identificar el número de marco de memoria?

2. ¿Qué organización de archivos deberá ser escogida para maximizar la eficiencia en términos de velocidad de acceso, uso del espacio de almacenamiento y facilidad de actualización (añadir, eliminar y modificar) cuando la data del archivo es:

- (a) Actualizada poco frecuente y accedida frecuentemente de manera aleatoria o directa. Explique
- (b) Actualizada frecuentemente y accedida frecuentemente de manera secuencial. Explique.
- (c) Actualizada frecuentemente y accedida frecuentemente de manera aleatoria o directa. Explique.

Dado el siguiente esquema de manejo de memoria



P= Bit Present (1: la página está en memoria principal, 0: la página no está en memoria principal)

El espacio de direcciones lógicas es de 512Kb y el espacio de direcciones físicas es de 64Kb. El tamaño de las páginas es de 1Kb. El número máximo de segmentos por proceso es 8.

- a) Escriba cómo, en general, una dirección lógica generada por el CPU es traducida a una dirección física utilizando un esquema por Segmentación Paginada.
- b) Indique el tamaño de cada uno de los siguientes campos en bits:

Segmento	
Página	
Desplazamiento	
Despl.	
Frame	

- c) Indique:

Número Máximo de entradas en la Tabla de Páginas

Número de marcos de memoria

Tamaño máximo de un segmento

- d) A qué dirección física, si existe, corresponden las siguientes direcciones virtuales. Puede dejar las direcciones expresadas en base 2.

(i) $13_{10} =$

(ii) $65549_{10} =$

(iii) $2052_{10} =$