

# Tema 5.2

## Sistema Operativos (SSOO)

### Gestión de memoria: paginación y segmentación

# Índice

- Tipo de memoria
- Paginación
- Segmentación

# Tipo de memoria

La memoria principal se encuentra dividida en dos partes:

- Sistema operativo residente (kernel)
- Zona para los procesos de usuario.

La zona para procesos de usuarios se encuentra dividida a su vez en varias particiones que se asignarán a los procesos.

Hay dos grandes formas de dividir (**particionar**) la memoria:

- de forma **estática** o
- de forma **dinámica**.

# Particionamiento

Existen varias formas de tratar (particionar) la memoria principal (RAM)

- 1) Particionamiento **estático**
- 2) Particionamiento **dinámico**
- 3) **Paginación** simple
- 4) **Segmentación** simple
- 5) **Memoria virtual paginada**
- 6) Memoria virtual **segmentada**.

# Particionamiento

- Particionamiento estático: las particiones se establecen en el momento de arranque del SO y permanecen fijas durante todo el tiempo.
- Particionamiento dinámico: las particiones cambian de acuerdo a los requisitos de los procesos.

## Particionamiento estático

- La memoria principal se divide en un conjunto de particiones estáticas de tamaño fijo, que no tienen por qué ser iguales.
- Un proceso se puede cargar en una partición de menor o igual tamaño.
  - Hueco – bloque de memoria disponible; hay huecos de diversos tamaños repartidos por toda la memoria.
  - Cuando llega un proceso se le asigna un hueco lo suficientemente grande para que quepa.

# Asignación de Memoria

El SO mantiene información sobre:

- a) particiones asignadas
- b) particiones libres (huecos).

Hay varias formas de satisfacer una solicitud de tamaño  $n$  partiendo de una lista de huecos:

- Primer ajuste (First-fit): Se asigna el primer hueco lo suficientemente grande
- Mejor ajuste (Best-fit): Se asigna el hueco más pequeño que es lo suficientemente grande; hay que buscar en la lista entera de huecos (salvo si está ordenada por tamaño)
- Desperdicia el menor espacio posible

# Asignación de Memoria

**Ventajas:** sencilla de implementar

**Desventajas:** Empleo ineficiente de la memoria debido a la aparición de la fragmentación interna.

Otra desventaja es que el número de procesos activos es fijo: habrá tantos procesos como particiones tenga la memoria.

Es decir, el número de procesos cargados en memoria dependerá de cuántas particiones podamos hacer según la cantidad de memoria que tengamos.

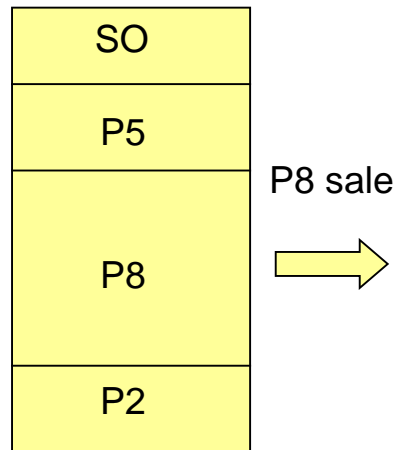
# Asignación de Memoria

## Fragmentación interna

- Si un proceso ocupa 3 KB, y el tamaño de la partición es de 4 KB, 1 KB se queda sin utilizar.

## Ejemplo:

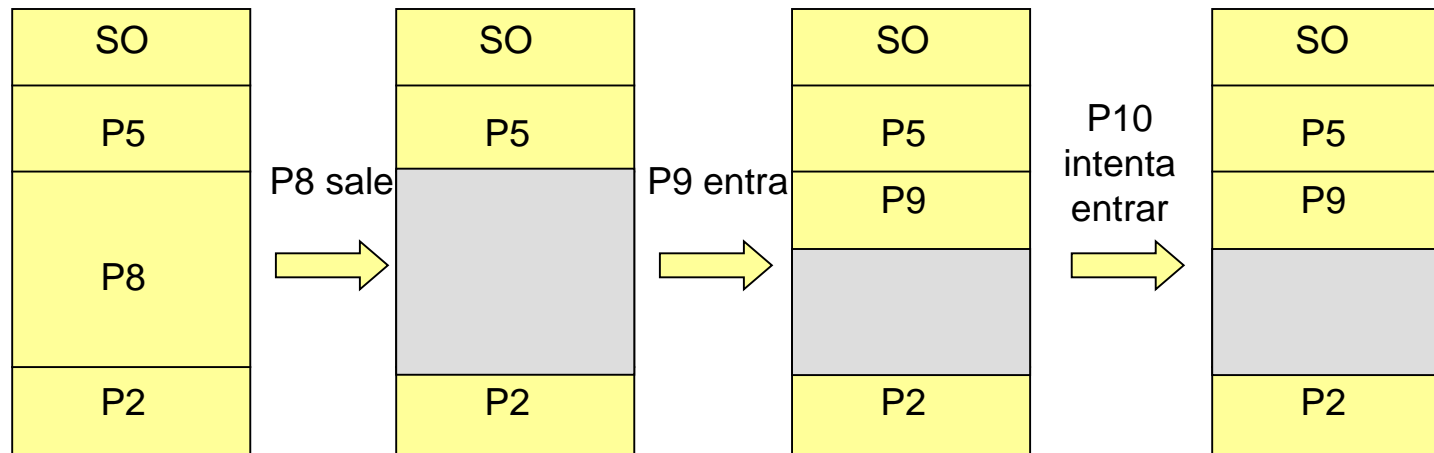
- Tengo hechas 4 particiones estáticas en la memoria  
Sale P8 (puede ser elegido para ejecutarse)





# Asignación de Memoria

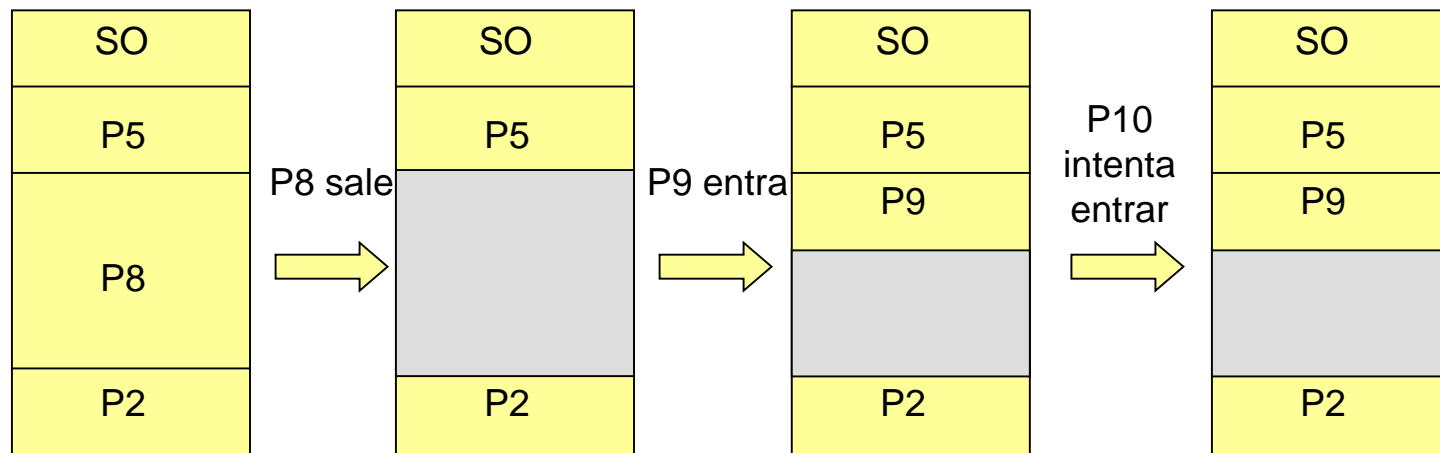
- Entra P9 ,que es más pequeño que P8, y se le asigna la partición que ha dejado libre P8
- Veamos lo que pasa luego cuando llega P10 y quiere entrar.



# Asignación de Memoria

## Fragmentación interna

- Como a cada partición le corresponde un proceso, aún pudiendo entrar por tamaño no puede.
- Se llama fragmentación interna porque está dentro de la partición ya hecha.



# Asignación de Memoria

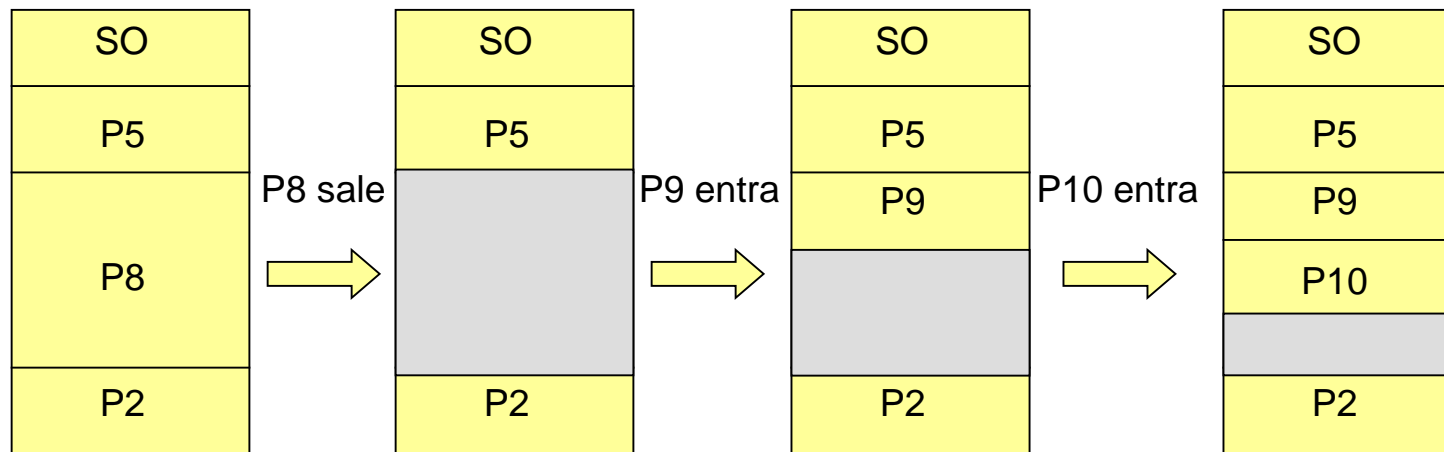
## Particionamiento dinámico

- Las particiones se crean dinámicamente, de forma que cada proceso se carga en una partición de exactamente el mismo tamaño que el proceso.
- Ahora el tamaño y ubicación de las particiones no es fijo sino que cambia a lo largo del tiempo.
- Cuando llega un proceso se le asigna memoria de un hueco lo suficientemente grande para que quepa.
- Con el espacio sobrante del hueco se crea una nueva partición libre (hueco).

# Asignación de Memoria

## Ventajas

- Ventajas: no hay fragmentación interna
- Uso más eficiente de la memoria principal
- Desventajas: Aparece la fragmentación externa (no está dentro de la partición, sino entre particiones) → ver hueco pequeño entre P10 y P2.



# Asignación de Memoria

## Ventajas:

- Ventajas: no hay fragmentación interna
- Uso más eficiente de la memoria principal

## Desventajas:

- Aparece la fragmentación externa (no está dentro de la partición, sino entre particiones) → ver hueco pequeño entre P10 y P2.
- Además se produce un uso ineficiente del procesador debido a la necesidad de **compactación** para contrarrestar la fragmentación externa.

# Asignación de Memoria

- La **compactación** es la reubicación de procesos para ocupar los huecos libres que van quedando.
- Es decir, los procesos se van reubicando en memoria.
- Alto coste computacional.

# Asignación de Memoria

## Paginación simple

- La memoria principal se divide en un conjunto de marcos de igual tamaño.
- Cada proceso se divide en una serie de páginas del mismo tamaño que los marcos.
- Un proceso se carga situando todas sus páginas en marcos libres pero no necesariamente contiguos.

**Ventajas:** no tiene fragmentación externa.

**Desventajas:** Hay una pequeña cantidad de fragmentación interna (puede haber páginas finales del proceso que ocupen menos que los marcos).

# Asignación de Memoria

## Segmentación simple

- Cada proceso se divide en una serie de segmentos. Un proceso se carga situando todos sus segmentos en particiones dinámicas que no tienen por qué ser contiguas.
- Es decir, es como la paginación, pero los marcos no son de tamaño fijo, sino dinámicos.

**Ventajas:** no tiene fragmentación interna.

**Desventajas:** tienen una pequeña fragmentación externa.



# Asignación de Memoria

## Paginación con memoria virtual

- Es igual que la paginación simple, solo que no hace falta cargar todas las páginas de un proceso en memoria principal.
- Las páginas no residentes que se necesiten se traerán de disco más tarde, de manera automática.
- Un proceso puede ser retirado temporalmente de la memoria a algún sistema de almacenamiento auxiliar, normalmente disco duro (memoria secundaria).
- Más tarde será incorporado de nuevo a la memoria para que continúe su ejecución.
- A este proceso se le llama paginación o intercambio (**swaping**).

# Asignación de Memoria

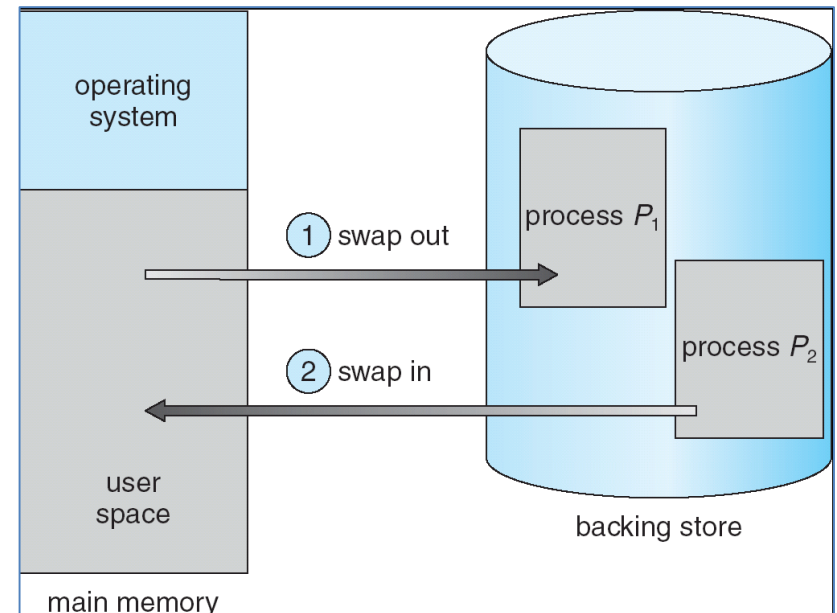
## Paginación con memoria virtual

- Almacenamiento auxiliar – disco rápido con capacidad suficiente para albergar copias de imágenes de memoria para todos los usuarios.
- Debe proporcionar acceso directo a estas imágenes de la memoria.
- A este espacio de almacenamiento auxiliar se le llama memoria virtual o memoria de intercambio (o swap).

# Asignación de Memoria

## Paginación con memoria virtual

- **Ventajas:** No hay fragmentación externa
- Gran espacio (aunque virtual) para el proceso.
- **Desventajas:** El acceso a la memoria de disco (secundaria) siempre es más lento.



# Asignación de Memoria

## Segmentación con memoria virtual

- Como la segmentación simple sólo que no hace falta cargar todas los segmentos de un proceso en memoria principal.
- Los segmentos no residentes que se necesiten se traerán de disco más tarde, de manera automática.

## Ventajas

- No hay fragmentación interna.
- Gran espacio (aunque virtual) para el proceso.

## Desventajas

- El acceso a la memoria de disco (secundaria) siempre es más lento.

# Paginación y Memoria física

- En la paginación se divide la memoria física en bloques de tamaño fijo llamados marcos.
- Se divide el proceso en páginas del mismo tamaño que los marcos.
- Se puede asignar memoria al proceso siempre que haya algún marco disponible (una página por marco), y no tienen por qué ser contiguo.
- Como en la paginación los marcos tienen un tamaño fijo, puede ser que parte de ese marco quede sin usar (típicamente la página final del proceso).
- Recordad que en este esquema aparece la fragmentación interna.

# Paginación y Memoria física

- Así, si el tamaño de marco (acorde con el de página) es muy grande es más probable que aparezca fragmentación interna
- El tamaño de página que se usa en Windows y en Linux es de 4 Kb.
- Esto es con la memoria física (o RAM) ¿Y la memoria virtual?
- La memoria virtual se implementa también sobre un esquema de paginación.
- De hecho en Windows se le llamó archivo de paginación (C:\pagefile.sys).
- En Linux es la partición swap que creamos en la instalación.

# Espacios de direcciones: físicas y lógicas

El concepto de espacio de direcciones lógicas vinculado a un espacio de direcciones físicas separado es crucial para una buena gestión de memoria.

- **Dirección lógica** – es la dirección que genera el proceso (en realidad el compilador); también se conoce como dirección virtual.
- **Dirección física** – dirección de la memoria que traduce la unidad de memoria a partir de la dirección lógica.

# Tabla de páginas

- Cuando un proceso es cargado en memoria, se cargan todas o parte de sus páginas en marcos libres y se completa su tabla de páginas.
- En la tabla de páginas de un proceso, se encuentra la ubicación de los marcos que contienen a cada una de sus páginas.
- La paginación de un programa la realiza el compilador y en ella cada dirección lógica se expresará mediante dos valores: Número de segmento (s) y desplazamiento dentro del segmento (d).
- Dicho de otra forma, las direcciones lógicas del proceso se forman como un número de página y de un desplazamiento dentro de esa página.



# Tabla de páginas

- Página ( $p$ ) – usado como índice en la tabla de páginas que contiene la dirección base de cada página en memoria física
- Desplazamiento ( $d$ ) – se combina con la dirección base para definir la dirección de memoria física que se envía a la unidad de memoria.
- El número de página es usado como un índice dentro de la tabla de páginas, y una vez obtenida la dirección del marco de memoria, se utiliza el desplazamiento para componer la dirección real o dirección física.
  - Este proceso se realiza por la MMU.
  - Es un proceso hardware y no software

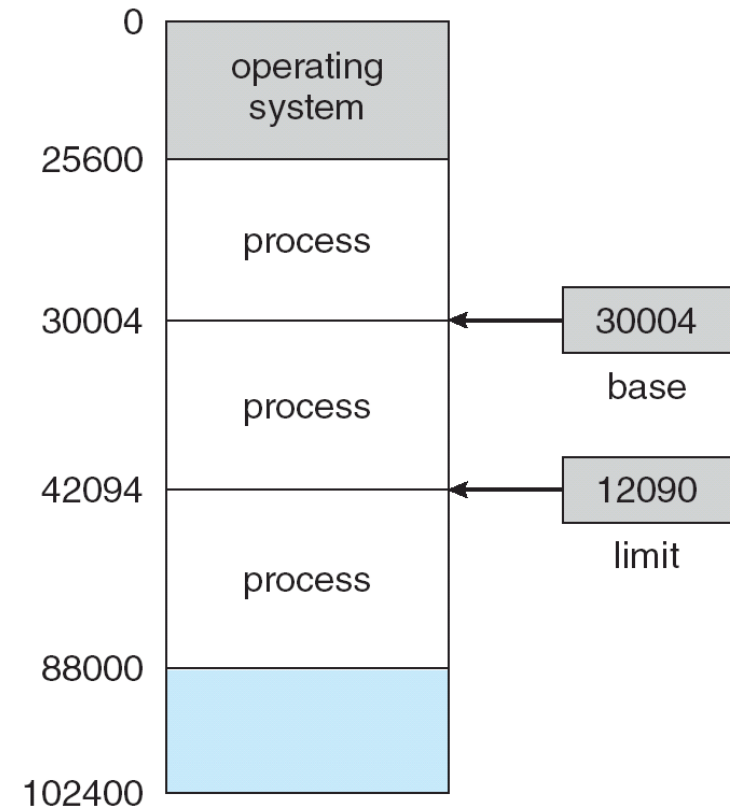
número de página	desplazamiento
$p$	$d$
$m - n$	$n$

# Tabla de páginas

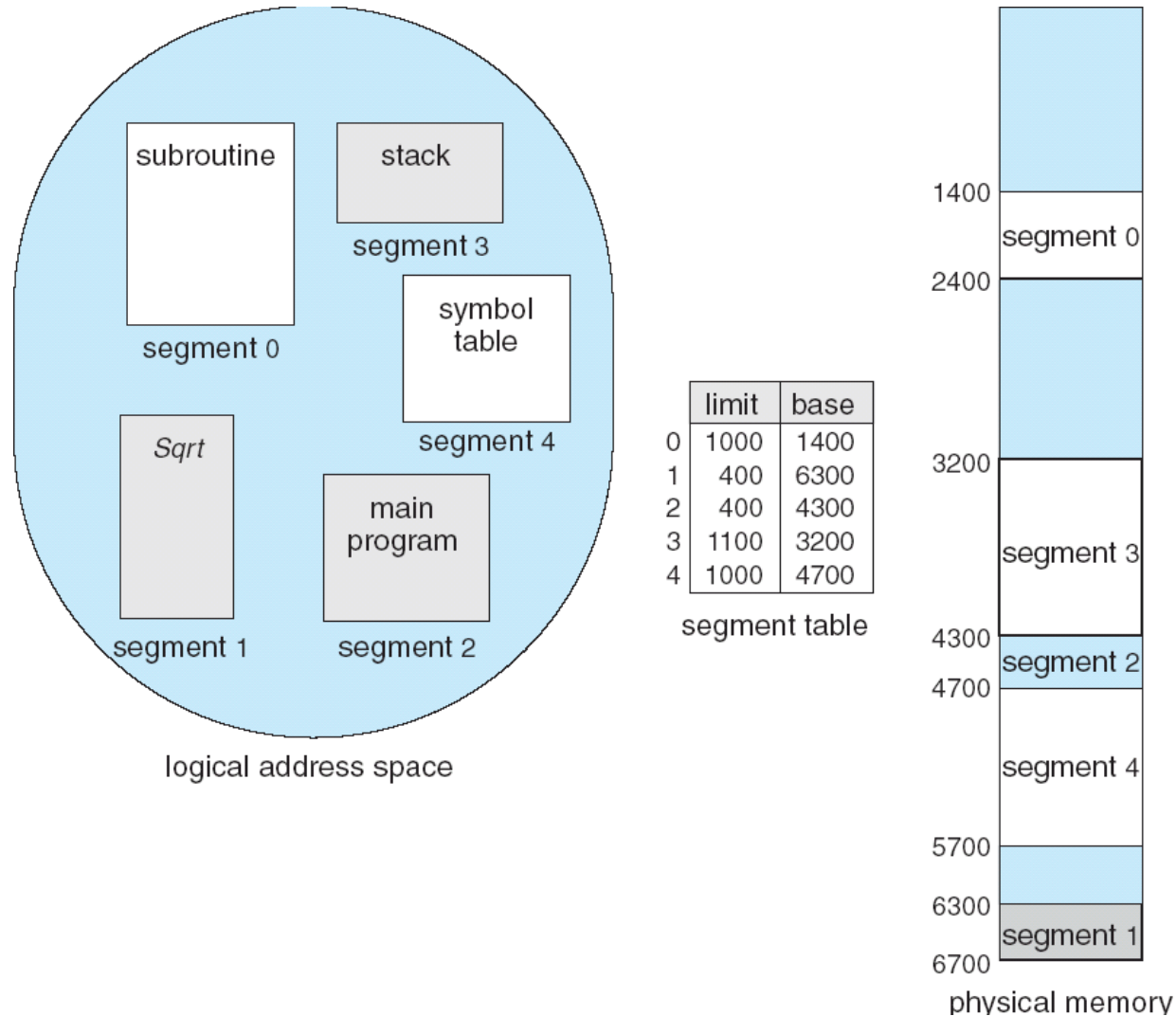
- Esto es en paginación, pero, ¿qué ocurre en segmentación?
- En segmentación los marcos no tienen el mismo tamaño y se crean de forma dinámica.
- Es necesario por tanto decir dónde empieza y dónde acaba el segmento.
- En lugar de una tabla de páginas tenemos una tabla de segmentos
- **Tabla de segmentos** – contiene información sobre la ubicación de los segmentos en memoria
- Cada entrada tiene:
  - **Registro base** – contiene la dirección física en la que comienza el segmento
  - **Registro límite** – especifica la longitud del segmento

# Tabla de páginas

- Así, un par de registros base y límite definen el espacio de direcciones físicas.
- La segmentación de un programa la realiza el compilador y en ella cada dirección lógica se expresará mediante dos valores: Número de segmento (s) y desplazamiento dentro del segmento (d)
- Lo mismo que en la paginación.



# Tabla de páginas



# Bibliografía

- **CARRETERO**, Jesús, **GARCÍA**, Félix, **DE MIGUEL**, Pedro, **PÉREZ**, Fernando. Sistemas Operativos: una visión aplicada. McGraw-Hill, 2001.
- **STALLINGS**, William. **Sistemas operativos: aspectos internos y principios de diseño. 5ª Edición. Editorial Pearson Educación. 2005. ISBN: 978-84-205-4462-5.**
- **TANENBAUM**, Andrew S. Sistemas operativos modernos. 3ª Edición. Editorial Prentice Hall. 2009. ISBN: 978-607- 442-046-3.

