

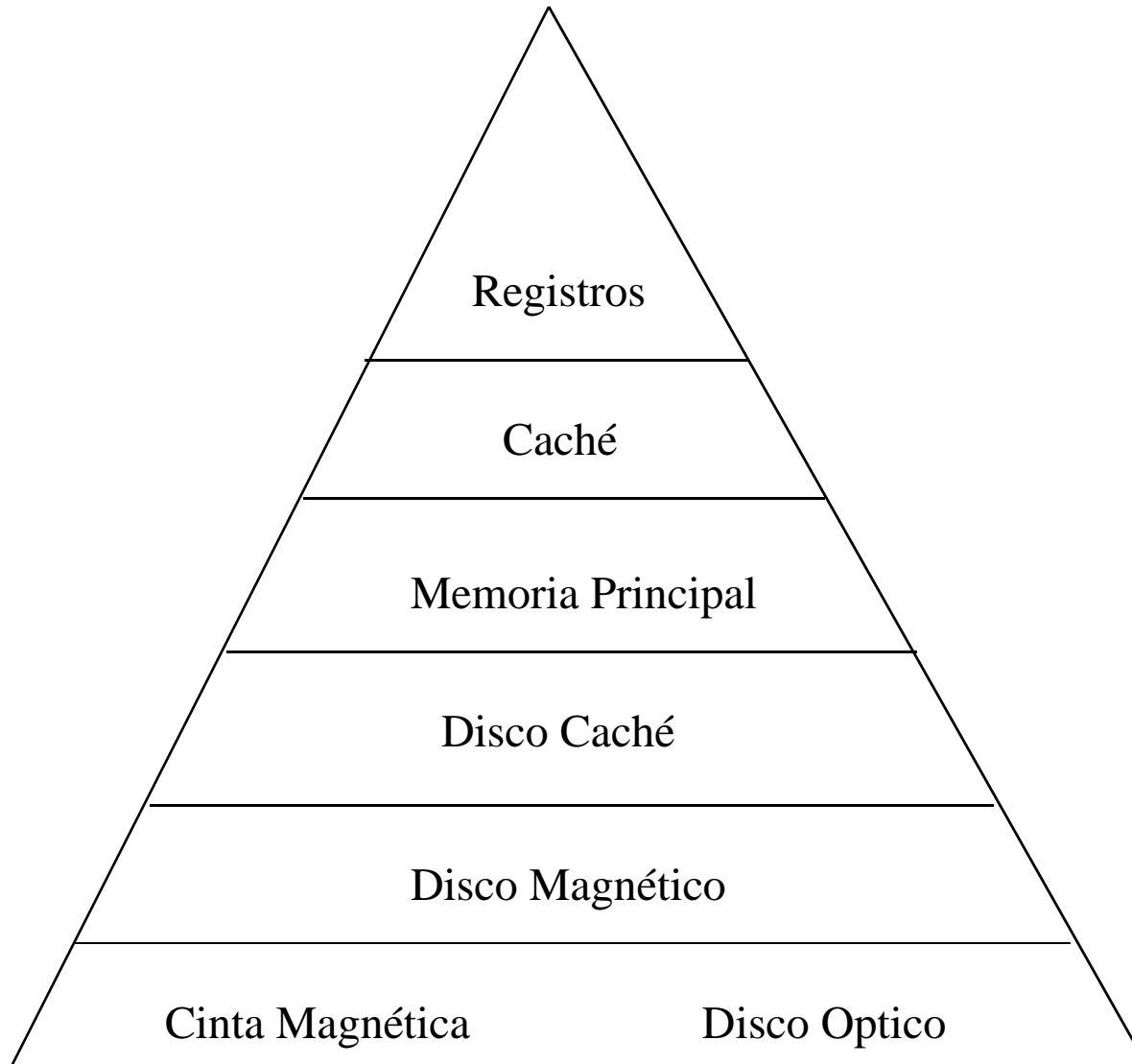
Administración de la memoria

aspectos fundamentales

La memoria

- Organización y administración de la memoria principal
- Memoria principal = memoria primaria = memoria real
- Memoria = almacenamiento

Jerarquía Memoria



Organización de la memoria

- La forma de considerar la memoria principal
 - Multiusuario o un solo usuario
 - Asignación de espacio a procesos concurrentes
 - Dividir memoria en fragmentos denominados particiones
 - Donde colocar cada trabajo:
 - en forma continua
 - en forma salteada

Estrategias administración memoria

- Determinan el comportamiento de una organización de memoria determinada
- Se dividen en:
 1. Estrategias de obtención
 - a) Estrategias de obtención por demanda
 - b) Estrategias de obtención anticipada
 2. Estrategias de colocación
 3. Estrategias de reemplazo

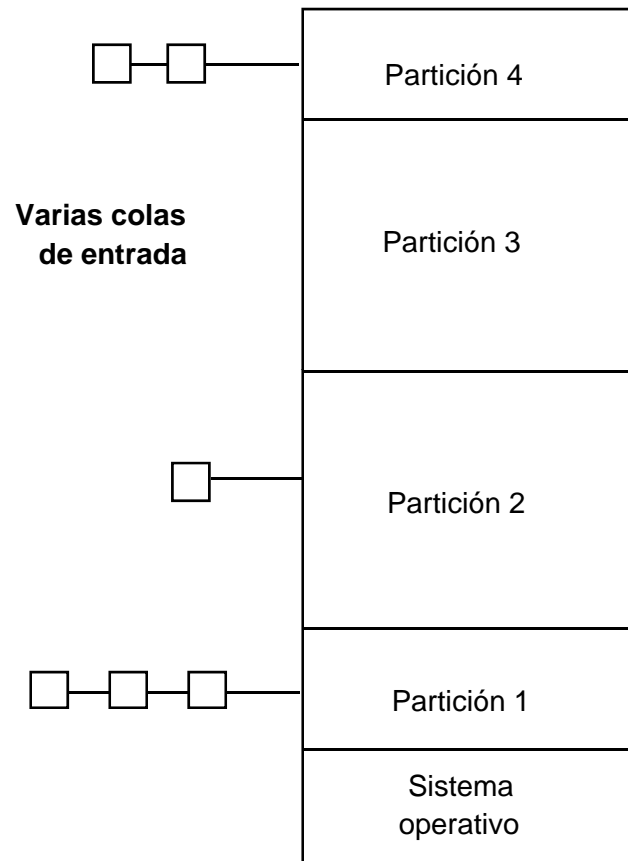
Asignación continua vs no-continua

- *Asignación memoria continua.*
 - cada programa debe ocupar bloques continuos de localidades de memoria
- *Asignación no continua*
 - programa es dividido en bloques o segmentos que pueden ser asignados a localidades no necesariamente continuas

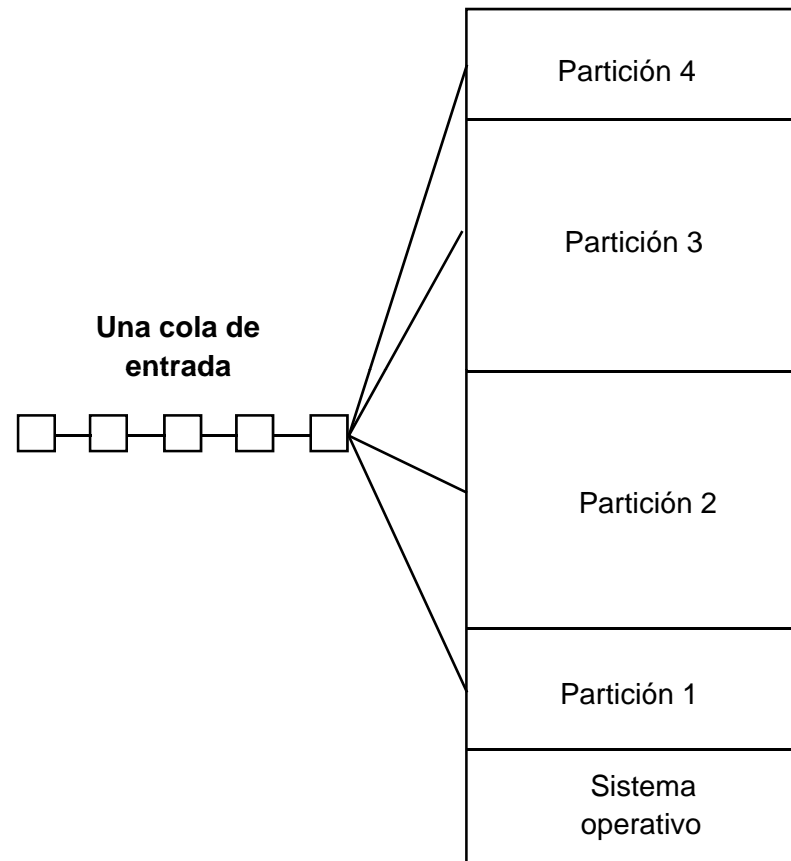
Multiprogramación y admon. memoria

- Facilita la programación de una aplicación al dividirla en dos o más procesos
- Capacidad de tener al mismo tiempo más de un proceso en memoria
- Poder proporcionar un servicio interactivo a varias personas al mismo tiempo
- Retardo atención procesos: dedican tiempo sustancial en la espera de la conclusión de E/S en disco.
- Dos tipos
 1. Multiprogramación con particiones fijas
 2. Multiprogramación con particiones variables

Colas múltiples vs colas únicas



(a)

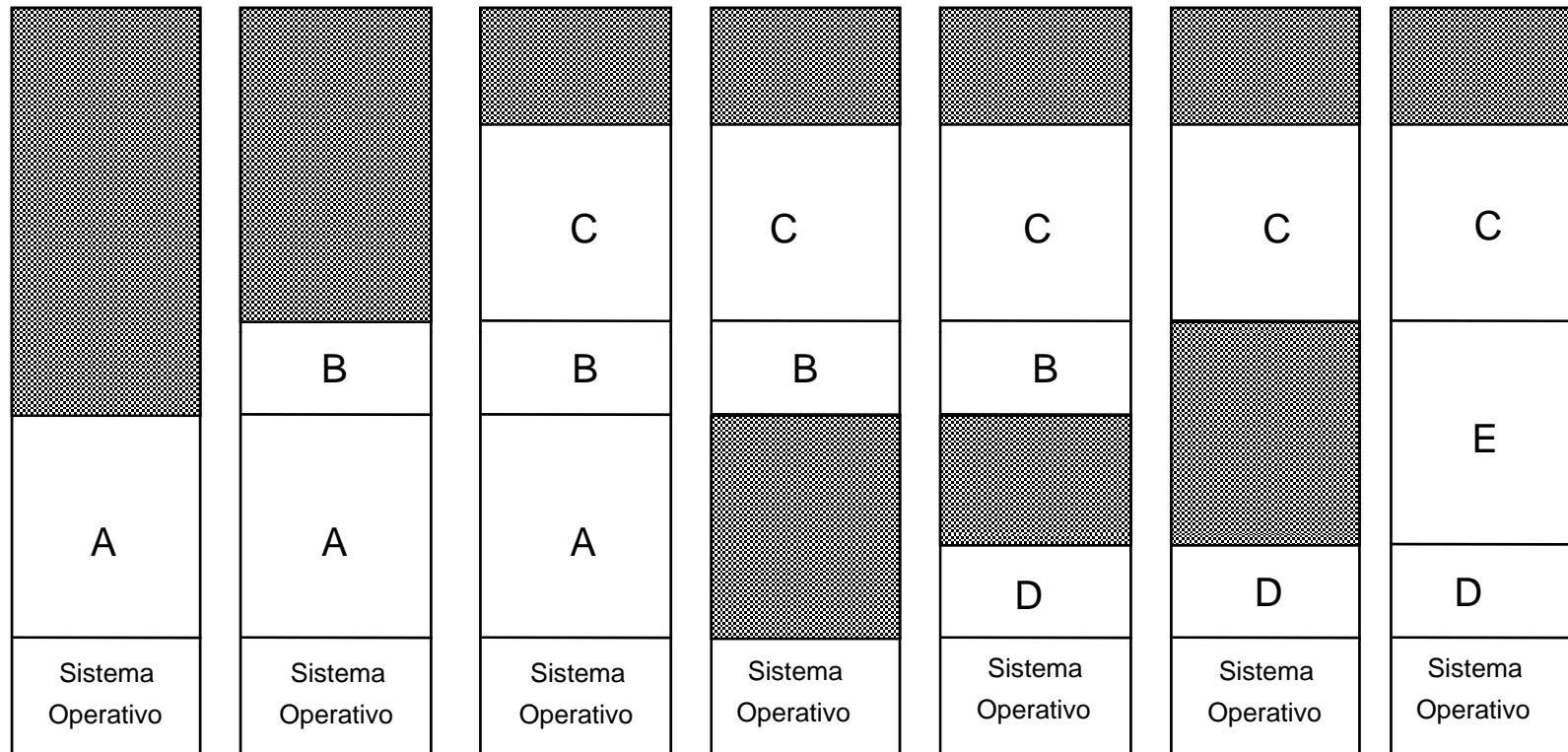


(b)

(a) Particiones fijas de memoria con colas de entrada independientes para cada partición

(b) Particiones fijas de la memoria, con una única colas de entrada

Particiones variables



Particiones variables: la asignación de memoria cambia cuando el proceso llega o sale de la memoria.



Memoria no utilizada

Protección multiprogramación

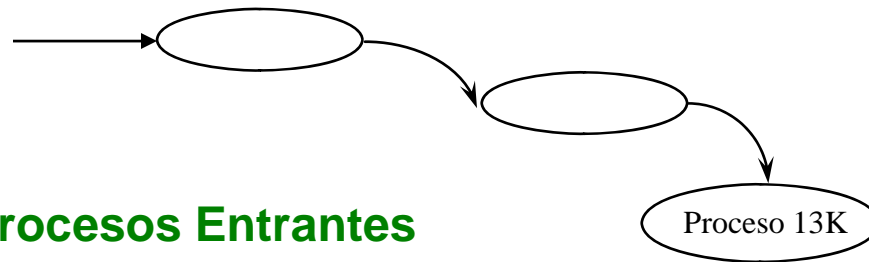
- Diversas tareas se ejecutan en direcciones distintas
- ¿Como asegurarse que un proceso no invada la dirección de otro?
- Tres posibles soluciones:
 1. Usar un offset por dirección/partición
 2. Dividir memoria en bloques y asignar códigos protección
 3. Usar registros especiales

Reemplazo o intercambio

- ***Intercambio:***
 - traslado de procesos de la memoria principal al disco y viceversa.
- ***Estrategias de reemplazo:***
 - usadas para determinar donde se va a ubicar el proceso dentro de la memoria principal
- ***Tipos estrategias:***
 1. First-fit
 2. Best-fit
 3. Worst-fit

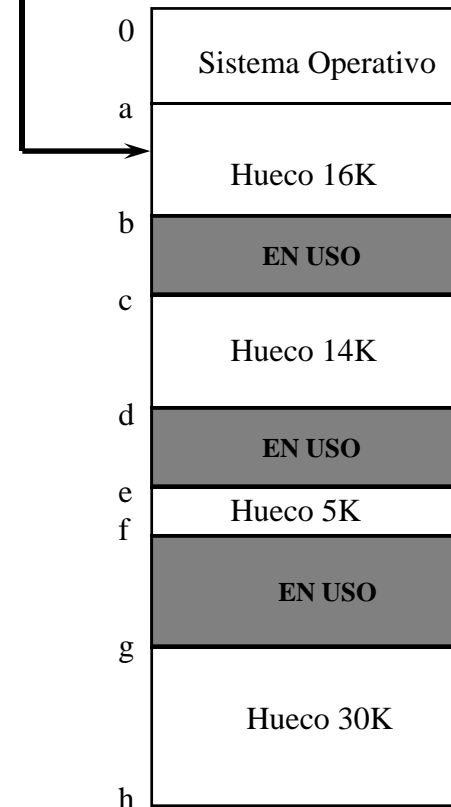
Estrategia First Fit

Lista Procesos Entrantes



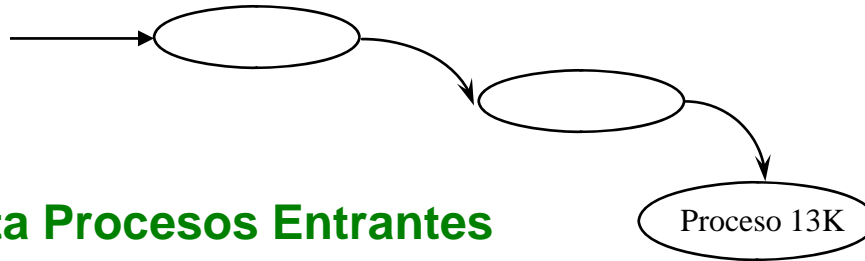
Lista Memoria Libre

Dirección Inicio	Tamaño
a	16K
c	14K
e	5K
g	30K



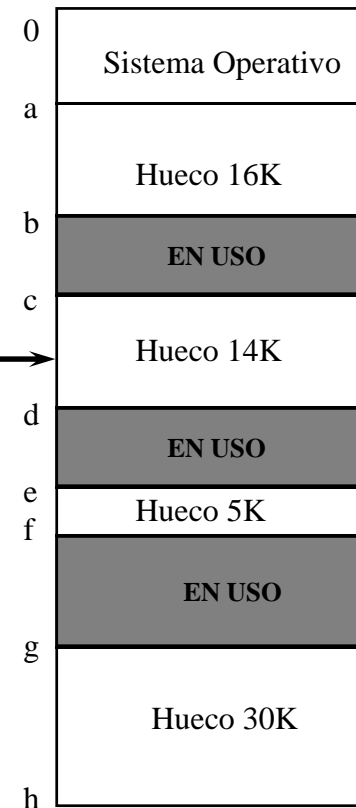
Estrategia Best-Fit

Lista Procesos Entrantes



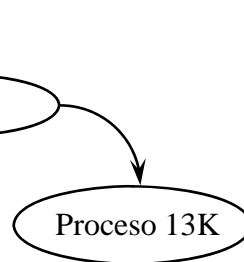
Lista Memoria Libre

Dirección Inicio	Tamaño
e	5K
c	14K
a	16K
g	30K



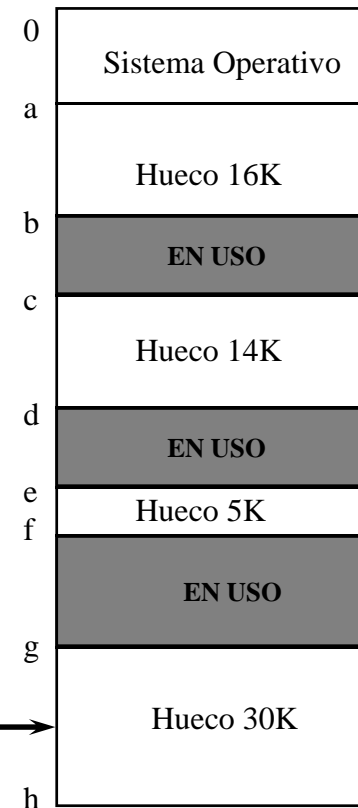
Estrategia Worst-Fit

Lista Procesos Entrantes



Lista Memoria Libre

Dirección Inicio	Tamaño
g	30K
a	16K
c	14K
e	5K



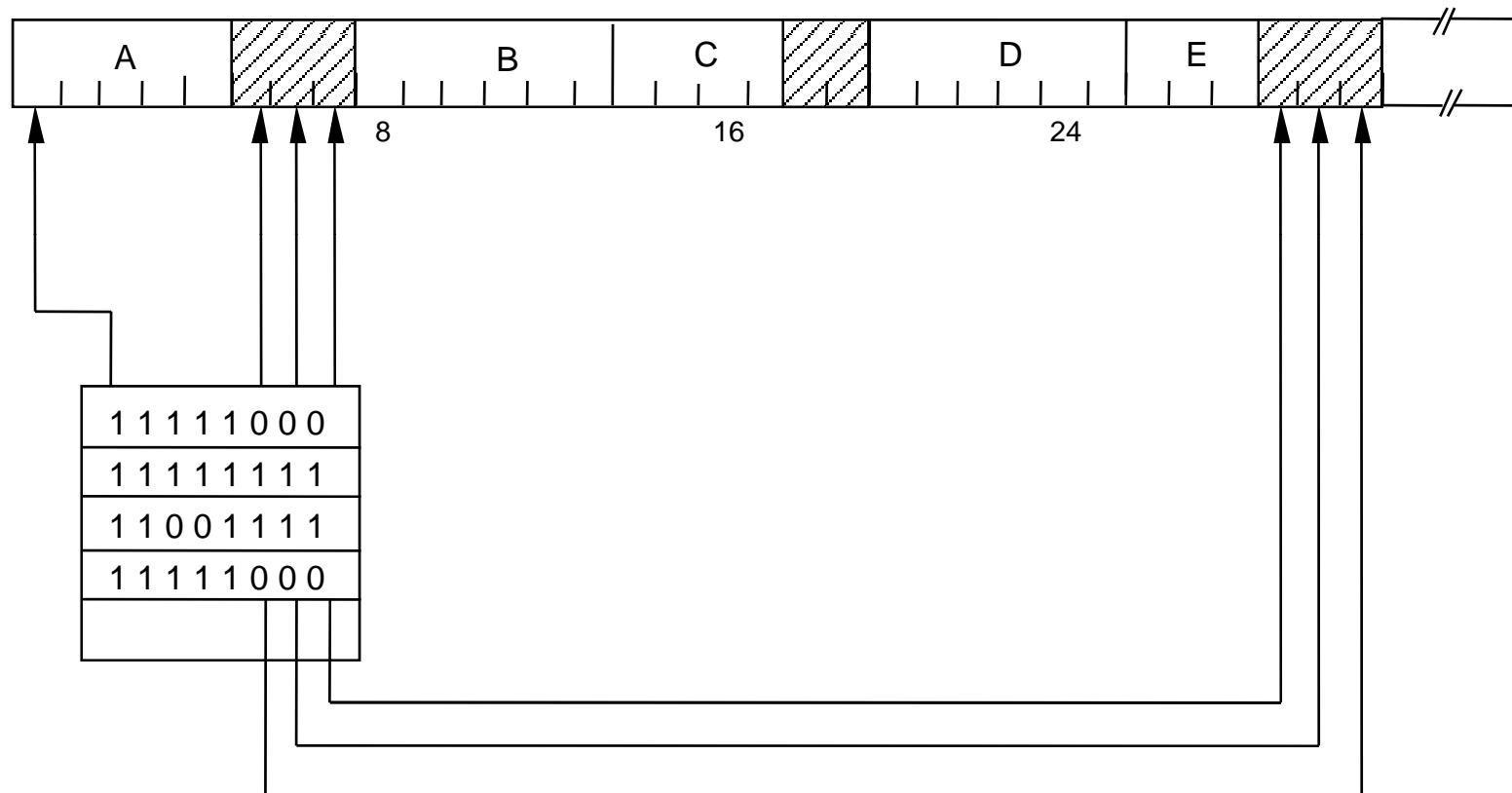
La fragmentación

- Número, posición y tamaño particiones varía en forma dinámica conforme el sistema es utilizado.
- Uso particiones variables provoca “huecos” en la memoria principal.
- Dos técnicas:
 1. Condensación de huecos
 2. Compactación del almacenamiento

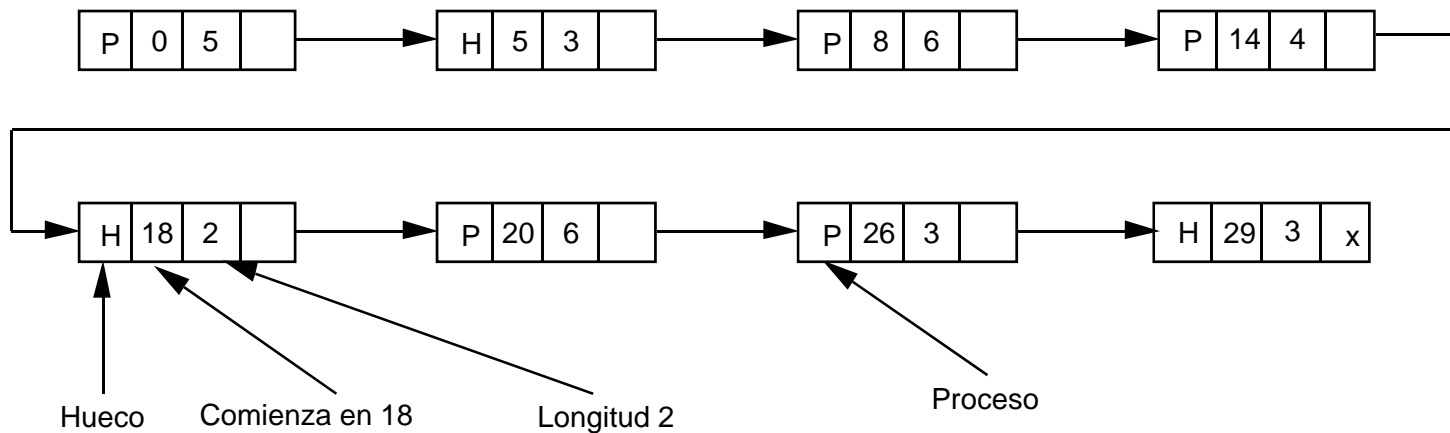
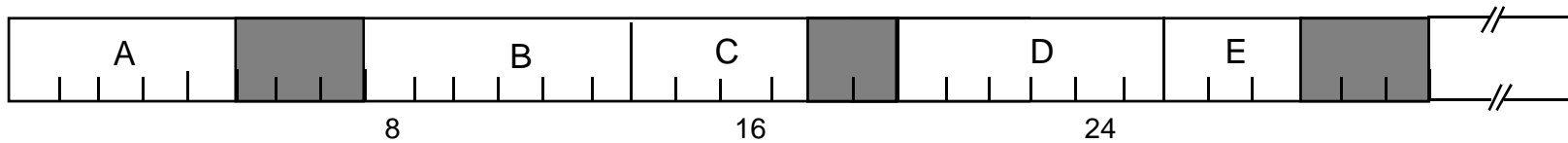
Administración particiones

- Mecanismo usado saber cuales particiones estan libres y cuales estan ocupadas.
- Existen tres mecanismos:
 1. Mapas de bits
 2. Listas ligadas
 3. Sistema de los asociados

Mapas de bits

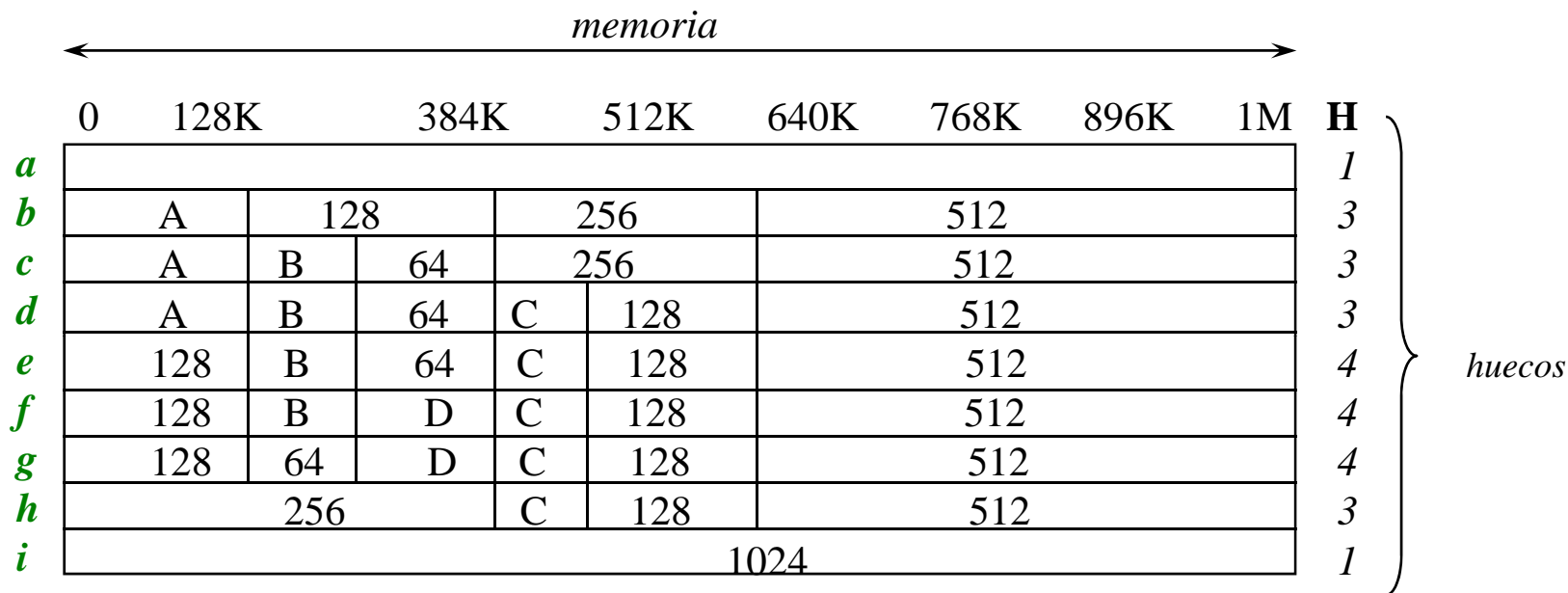


Listas ligadas



Una parte de la memoria con cinco procesos y tres huecos con la información en forma de lista ligada

Sistemas de asociados



a Al principio, (no se ha hecho ninguna solicitud)

b Solicitud de 70

c Solicitud de 35

d Solicitud de 85

e Regreso de A

f Solicitud de 60

g Regreso de B

h Regreso de D

i Regreso de C

Ventajas Sistemas Asociados

- Cuando se libera un bloque de tamaño 2^k , el administrador debe buscar en la lista de huecos de 2^k para ver si es posible una fusión.
- En algoritmos que dividen los bloques en forma arbitraria, la búsqueda se hace en todos los bloques

Desventajas sistemas asociados

- Ineficiente en terminos del uso de memoria.
- Todas las solicitudes deben redondearse a una potencia de dos
- Proceso 35K necesita un hueco de 64, por lo que se desperdician 29K
- Fragmentación interna
 - memoria desperdiciada dentro segmentos asignados (también conocida como slack space)
- Fragmentación externa:
 - huecos entre los segmentos, pero no existe hueco desperdiciado dentro de ellos (también conocida como checkerborading)