

Curso: Sistemas Operativos

3. Administración de memoria

(Parte II)

M.C. Laura Sandoval Montaño

3 Administración de Memoria

3.4 Asignación Contigua por Particiones Variables

Recordemos que los tipos de asignación contigua de memoria con multiprogramación son:

- De particiones fijas
- De particiones variables

Hasta el momento hemos estudiado la primera, ahora conoceremos y trabajaremos con la asignación contigua con multiprogramación que ocupa particiones variables.

Multiprogramación con particiones variables

Funcionamiento:

- La situación inicial, antes de la asignación de espacio a procesos del usuario, es una sola partición del tamaño del área para el usuario.
- Conforme van llegando los procesos, se les asigna memoria contigua al tamaño requerido, hasta que ya no quepa otro proceso más; el cual se ingresa a una cola de espera.
- Cuando los procesos van terminando, liberan el espacio asignado. A dicho espacio liberado se le llama **hueco** o **fragmento externo**.
- Cuando se crean los huecos, se revisa si hay procesos en cola de espera, y se les va asignando un hueco, en donde quepa el proceso completo.

Multiprogramación con particiones Variables.

Funcionamiento (continuación):

- Existen varias **técnicas de asignación de huecos** a los procesos que están en cola de espera o, si ésta está vacía, los que vayan llegando.
- Si el proceso, al que se le asigna un hueco, no cubre todo el hueco, se genera otro hueco con el espacio no utilizado, de tal manera que otro proceso pueda utilizarlo si cabe completo en él.
- Así se van generando huecos cada vez más chicos, propiciando que no sean del tamaño adecuado para que entren completos otros procesos.
- Es entonces, que se aplican **técnicas de tratamiento de huecos** para obtener huecos suficientemente grandes de memoria para dar cabida a los procesos nuevos.

Antes de revisar, tanto las técnicas de asignación de huecos como de tratamiento de huecos, realicemos un ejercicio de asignación inicial de memoria utilizando Multiprogramación con particiones variables.

3 Administración de Memoria

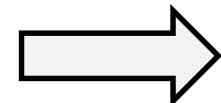
3.4 Asignación Contigua por Particiones Variables

Multiprogramación con particiones Variables.

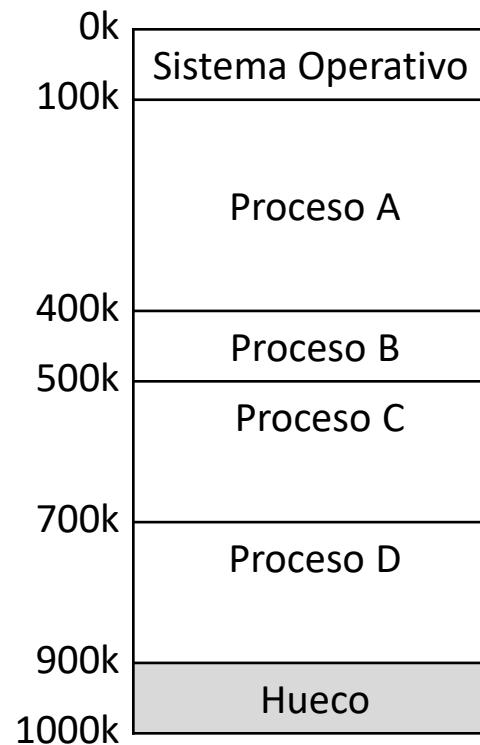
Ejercicio 1.

Un cierto sistema maneja memoria contigua con multiprogramación y particiones variables; el tamaño de su memoria es de 1,000 K, las direcciones más bajas se ocupan para almacenar al sistema operativo de tamaño 100 K. Si inicialmente la memoria del usuario está vacía, muestra cómo asigna espacio de memoria a los siguientes procesos de acuerdo a su orden de llegada:

Proceso	Tamaño (K)
A	300
B	100
C	200
D	200



asignación

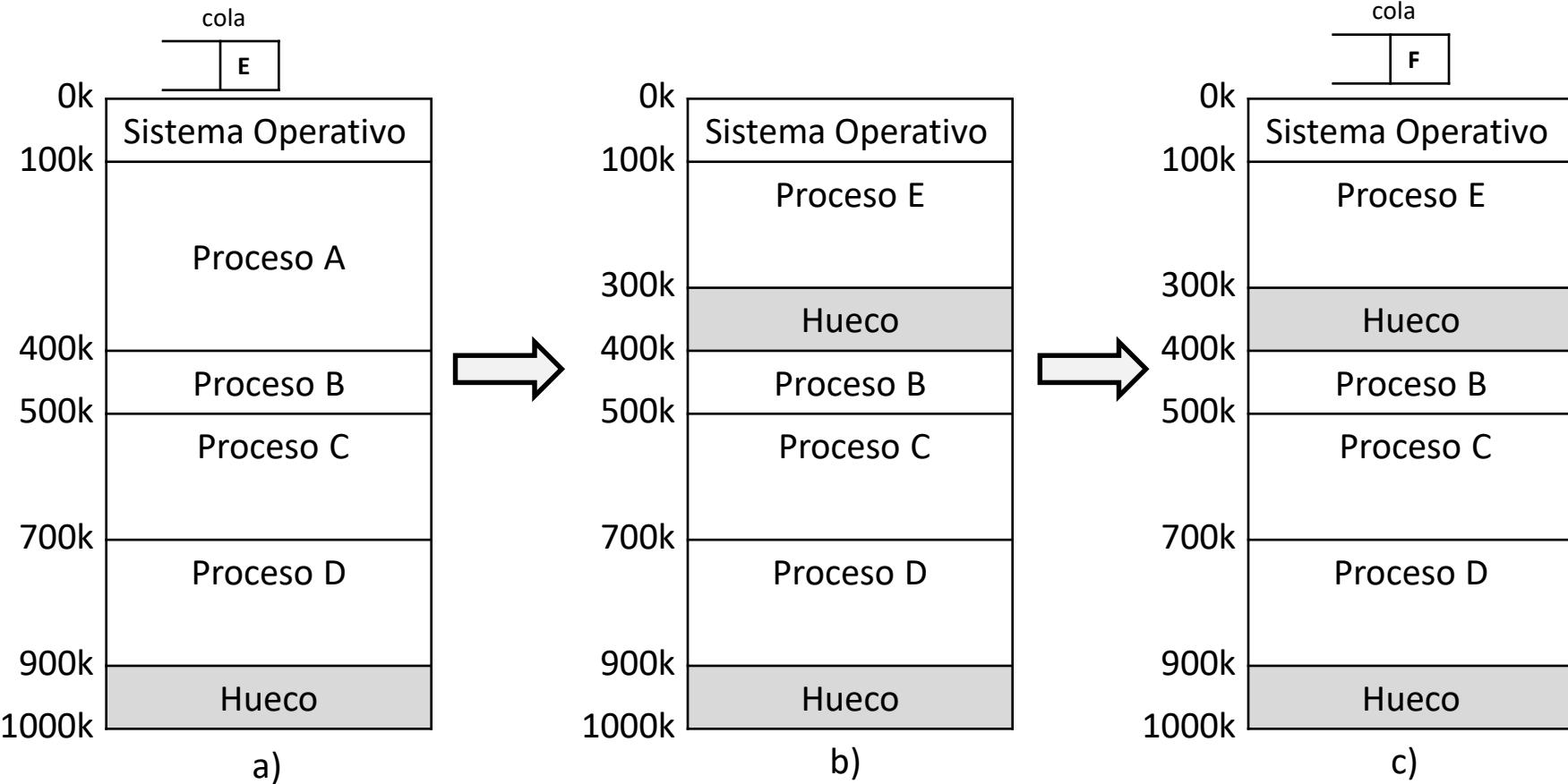


Multiprogramación con particiones Variables.

Ejercicio 2.

Considerando la situación de memoria final del sistema del Ejercicio 1, muestra cómo se haría la asignación de memoria con la siguiente secuencia de sucesos:

- a) Llega proceso E(200k), b) termina proceso A, c) llega proceso F(200k).



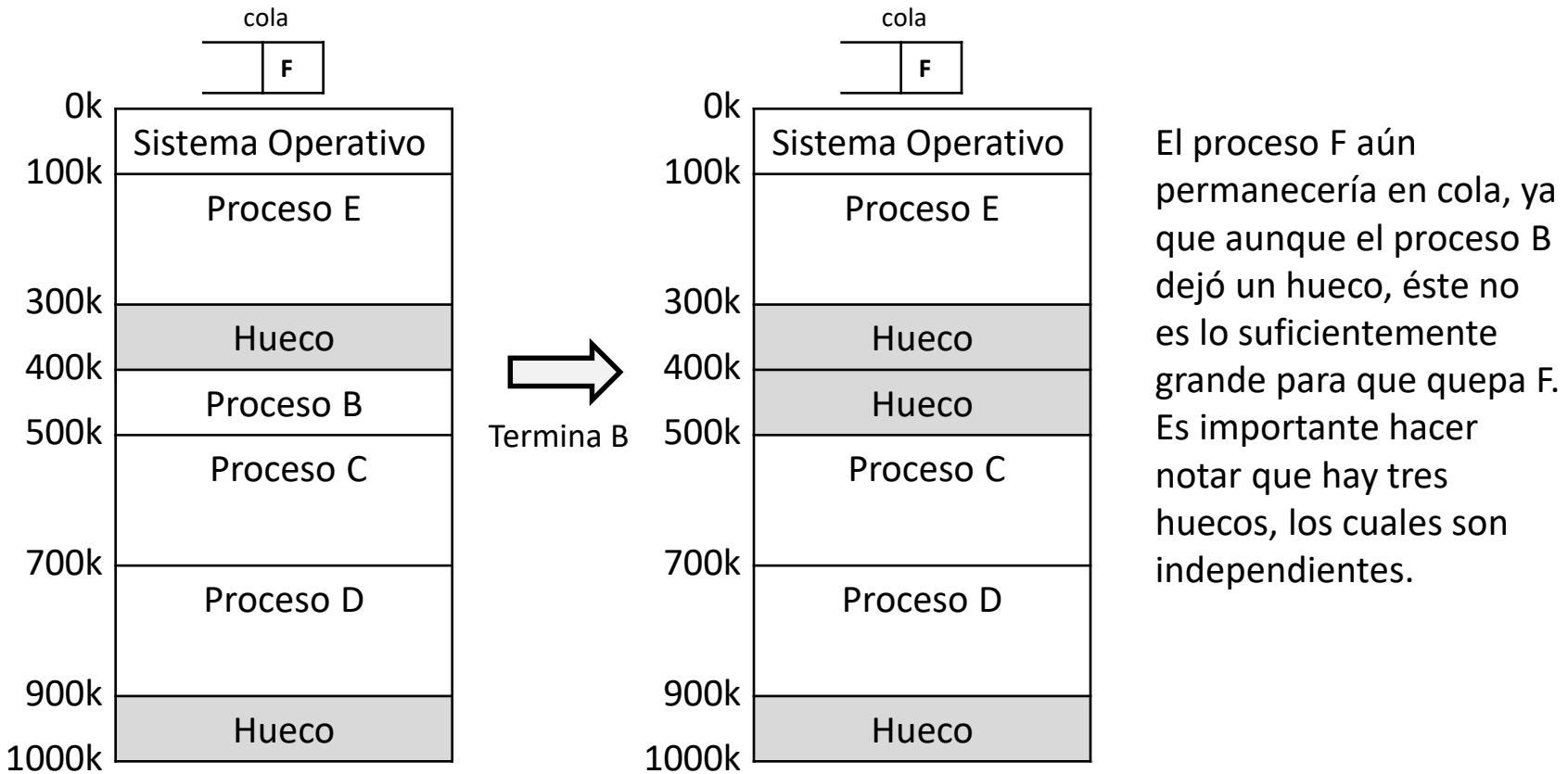
3 Administración de Memoria

3.4 Asignación Contigua por Particiones Variables

Multiprogramación con particiones Variables.

Si observamos la situación final de la memoria del Ejercicio 2, el proceso F no puede entrar a la memoria debido a que los dos huecos son de menor tamaño que el del proceso.

¿Qué sucedería si el proceso B terminara?



Multiprogramación con particiones Variables.

Técnicas de tratamiento de huecos.

Estas técnicas permitirán reorganizar los espacios de memoria disponibles (huecos) para que a los procesos que vayan llegando o estén en cola, se les pueda asignar memoria.

Condensación: Técnica que une huecos contiguos. Técnica sencilla porque sólo se define un hueco con la dirección inicial del primer hueco contiguo y la dirección final la del último hueco contiguo.

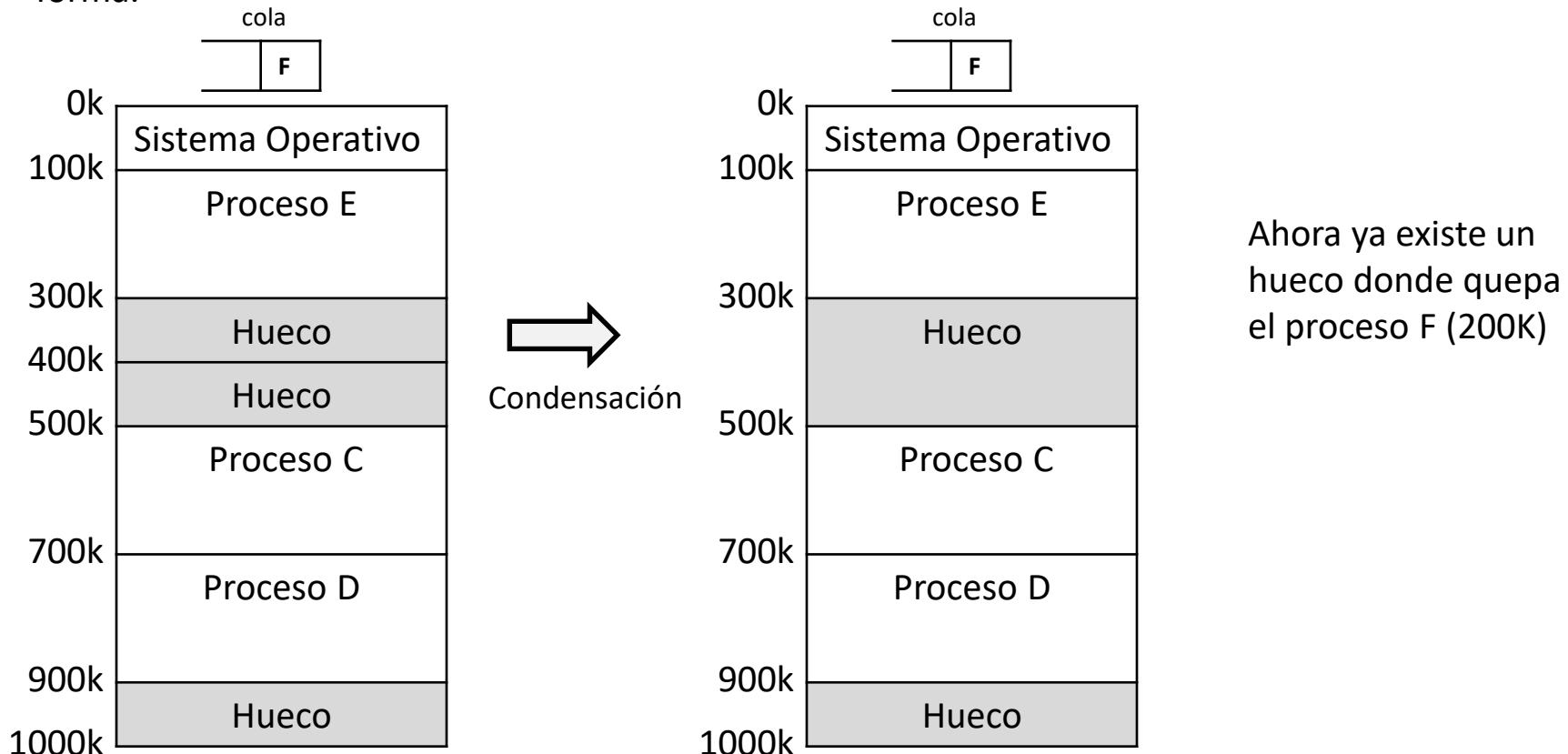
Compactación: Técnica que une todos los huecos (contiguos y dispersos) en un solo hueco. Técnica más complicada ya que, dependiendo dónde quede el hueco resultante, hay que reubicar a los procesos y sus datos. Consume tiempo de procesador.

Multiprogramación con particiones Variables.

Técnicas de tratamiento de huecos.

Ejemplo de Condensación:

De la situación última de memoria del ejercicio anterior, aplicamos condensación, de la siguiente forma.

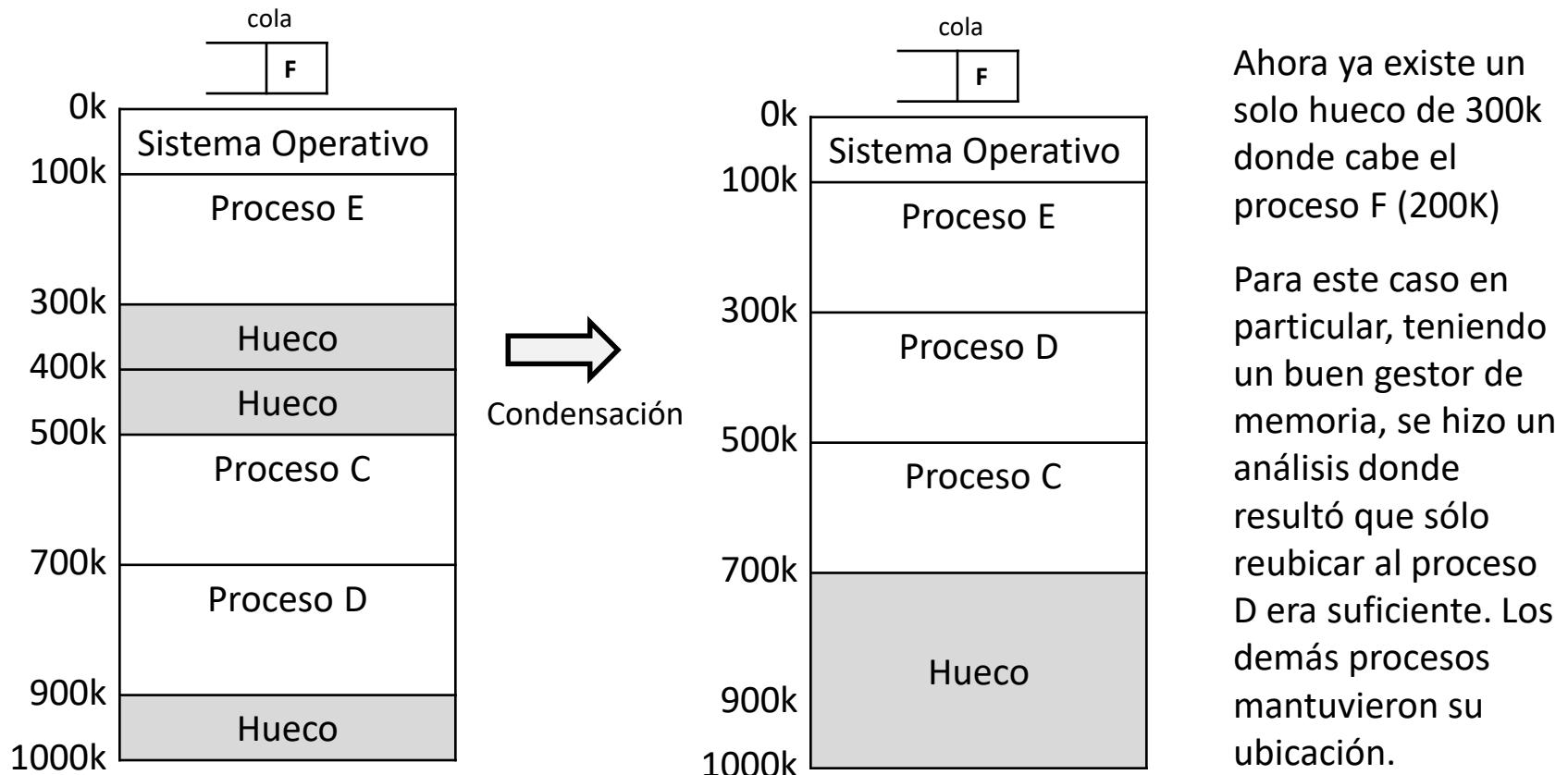


Multiprogramación con particiones Variables.

Técnicas de tratamiento de huecos.

Ejemplo de Compactación:

Ahora veamos cuál es la situación resultante si se aplica la Compactación:



Ahora ya existe un solo hueco de 300k donde cabe el proceso F (200K)

Para este caso en particular, teniendo un buen gestor de memoria, se hizo un análisis donde resultó que sólo reubicar al proceso D era suficiente. Los demás procesos mantuvieron su ubicación.

Multiprogramación con particiones Variables.

Técnicas de asignación de huecos

Estas técnicas deciden qué hueco será asignado al proceso que solicita memoria. Cada técnica aplica una estrategia. Existen tres estrategias:

- **Mejor ajuste.** Al proceso se le asigna el hueco donde quepa casi exacto. Esto con la intención de que el proceso no deje mucho espacio sin ocupar y la memoria del usuario esté usada casi al 100%.
- **Primer ajuste.** Al proceso se le asigna el primer hueco donde quepa. Esto con la intención de que no se emplee tiempo de procesador en calcular dónde colocar al proceso.
- **Peor ajuste.** Al proceso se le asigna el hueco más grande. Esto con la intención de dejar un espacio disponible lo suficientemente grande para que entre otro proceso.

¿Cuál estrategia considerarías la más conveniente y porqué?

Multiprogramación con particiones Variables.

Técnicas de asignación de huecos

Algunas desventajas:

Mejor ajuste. De manera pronta, van quedando muchos huecos muy pequeños en donde no cupieran más procesos. Utiliza más memoria y procesamiento para su administración ya que requiere mantener una tabla de ubicación de huecos ordenados por tamaño de menor a mayor.

Primer ajuste. Se van generando muchas particiones en las direcciones más bajas de memoria, lo que se van formando huecos muy pequeños. Entonces la búsqueda secuencial de huecos de tamaño suficiente para que entre un proceso va siendo más tardado.

Peor ajuste. Se agotan los huecos grandes, luego los medianos, quedando sólo pequeños. Utiliza más memoria y procesamiento para su administración ya que requiere mantener una tabla de ubicación de huecos ordenados por tamaño de mayor a menor.

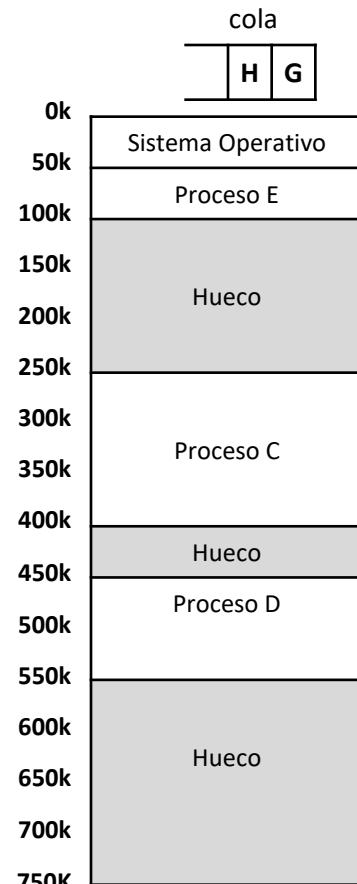
3 Administración de Memoria

3.4 Asignación Contigua por Particiones Variables

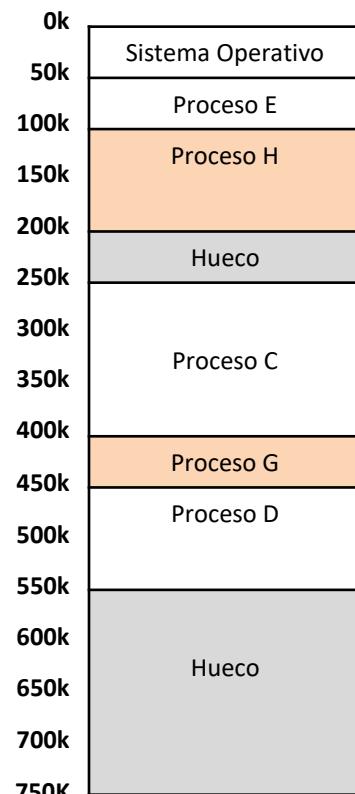
Multiprogramación con particiones Variables.

Ejercicio 3.

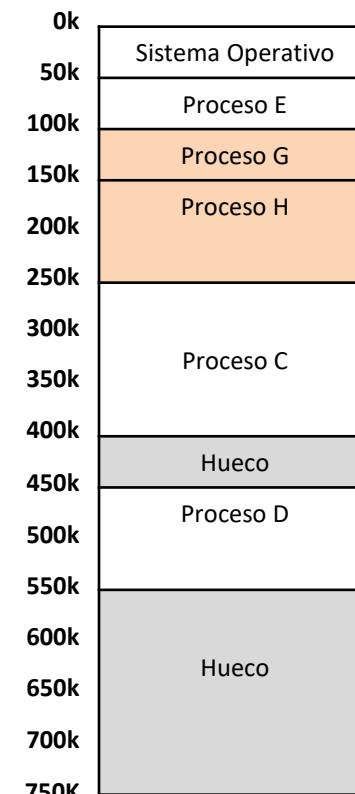
Partiendo de la situación de memoria a), muestra cómo se haría la asignación de huecos a los procesos G(50k) y H(100k), con cada una de las estrategias de asignación de huecos mencionadas.



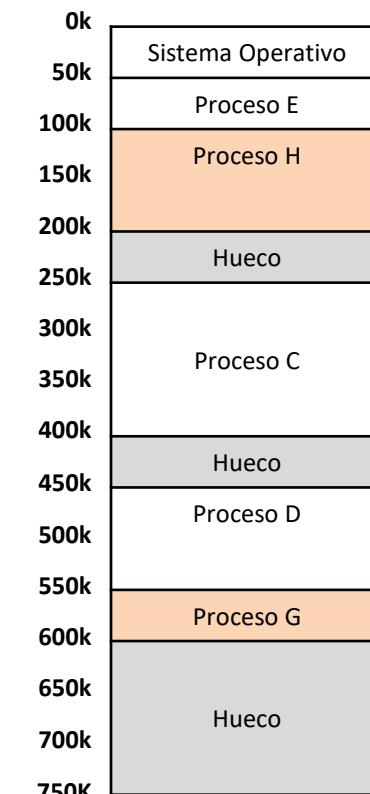
a) Estado inicial



b) Mejor ajuste



c) Primer ajuste



d) Peor ajuste

3 Administración de Memoria
3.4 Asignación Contigua por Particiones Variables

Multiprogramación con particiones Variables.

Preguntas sobre asignación contigua con multiprogramación y particiones variables. Son preguntas de tarea y de tipo examen.

Ejercicio tarea 1.

Un cierto sistema maneja memoria contigua con multiprogramación y particiones variables; el tamaño de su memoria es de 1,500 K, las direcciones más bajas se ocupan para almacenar al sistema operativo de tamaño 200 K. Si inicialmente la memoria del usuario está vacía, muestra cómo asigna espacio de memoria a los siguientes procesos de acuerdo a su orden de llegada (puedes presentar el esquema final nada más):

Proceso	Tamaño (K)
A	300
B	50
C	150
D	200
E	100
F	250
G	100

Multiprogramación con particiones Variables.

Ejercicio tarea 2.

Partiendo de la situación final de memoria del Ejercicio tarea 1, muestra cómo queda la memoria cuando se presentan los siguientes sucesos, para cada estrategia de asignación de huecos.

Proceso	Suceso
C	termina
F	termina
H(50)	llega
D	termina
I(100)	llega
J(200)	llega
A	termina
K(50)	llega

Multiprogramación con particiones Variables.

Ejercicio tarea 3.

Realiza un análisis comparativo entre las técnicas de tratamiento de huecos (condensación y compactación), en cuanto a:

- i) rapidez,
- ii) eficacia,
- iii) cuándo es conveniente aplicar una o aplicar la otra,
- iv) consumo de tiempo de procesador,
- v) otra (la que propongás).

Preséntalo en una tabla.

Ejercicio tarea 4.

Menciona dos diferencias sustanciales entre fragmento, generado en particiones fijas, y hueco, generado en particiones variables.