

Gestor de Procesos

Simulador Planificador de procesos



El proyecto consta de 3 fases:

- 1) Generación de procesos
- 2) Asignación de memoria
- 3) Planificación de procesos

1. Generación de procesos

Implementar una interfaz (consola) para la captura de datos de los procesos con los cuales se simulará el planificador.

i) Datos de los procesos:

a) **Id del proceso** (numérico o alfanumérico)

Es un identificador del proceso que lo hace único dentro del Sistema Operativo, por lo que no debe duplicarse.

b) **Nombre del proceso**

c) **Tamaño del proceso**

d) **Tiempo que requiere para su ejecución**

Recuerda que los procesos están huecos, con este tiempo simularemos la ejecución de instrucciones en el procesador.

e) **Prioridad del proceso** (numérico)

f) **Tiempo que requiere para operaciones de E/S** ^{**}(no se modela para éste semestre)

g) **Tiempo de llegada del proceso.**

Es el tiempo en el que se crea el proceso una vez iniciado el simulador. ^{**}(para este semestre los procesos SI pueden llegar en DIFERENTES TIEMPOS).

Todos los tiempos se simularán en milisegundos

- ii) La cola de procesos listos se implementará con una lista dinámica de nodos. **No deberá de usar las colecciones de java (TAD).**
- iii) Conforme se crean los procesos se van insertando en esta lista y conforme se van planificando se eliminarán de la cola. Por **cada cambio** (insertar o borrar) que se realice en la estructura cola de procesos listos **se deberá de imprimir su nuevo contenido.**



2. Asignación dinámica de memoria

Cuando un proceso se planifica se carga en la memoria y se crea dinámicamente su partición. Pero, para el simulador que se implementará **NO** se simulará la partición dinámica de la memoria, bastará con validar si hay espacio suficiente para cargar el proceso (tamaño restante de memoria – tamaño del proceso a planificar), y presentar en pantalla el espacio restante de la memoria una vez que se cargó el proceso.

Se cargarán los procesos en la memoria hasta que se llene o ya no exista espacio suficiente para almacenar el siguiente proceso en la cola de listos. Para ello, se deberá de implementar una cola FIFO de **procesos listos para ejecución**. Los procesos almacenados en esta cola son los que subirán a la CPU.

Salida a pantalla:

- Por cada cambio que exista en esta lista deberá de imprimir su contenido (cada vez que se inserte o borre un proceso se deberá de imprimir la lista).
- Cada vez que se libere espacio en memoria se deberá de imprimir el total de espacio disponible.

Ejemplo: Suponga un tamaño de memoria disponible para el usuario de 1024k y los siguiente tamaños de los procesos: p1=200k, p2 = 300k, p3 = 150k, p4=500k, p5=100k y p6=350k:

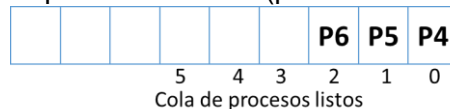
- ✓ El simulador deberá presentar a pantalla la siguiente información conforme se van simulando la carga de proceso a la memoria:

Subió el proceso P1 y restan 824 unidades de memoria
Subió el proceso P2 y restan 524 unidades de memoria
Subió el proceso P3 y restan 374 unidades de memoria

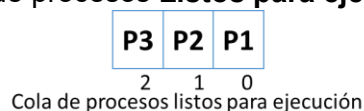
Los procesos P4, P5 y P6 no son cargados a memoria, debido a que P4 es el siguiente en subir por FIFO pero, no hay memoria suficiente.

- ✓ El simulador presentará a pantalla la información de las colas de procesos:

A. Cola de procesos **listos**: (procesos en espera para subir a la memoria)



B. Cola de procesos **Listos para ejecución**: (procesos en memoria)

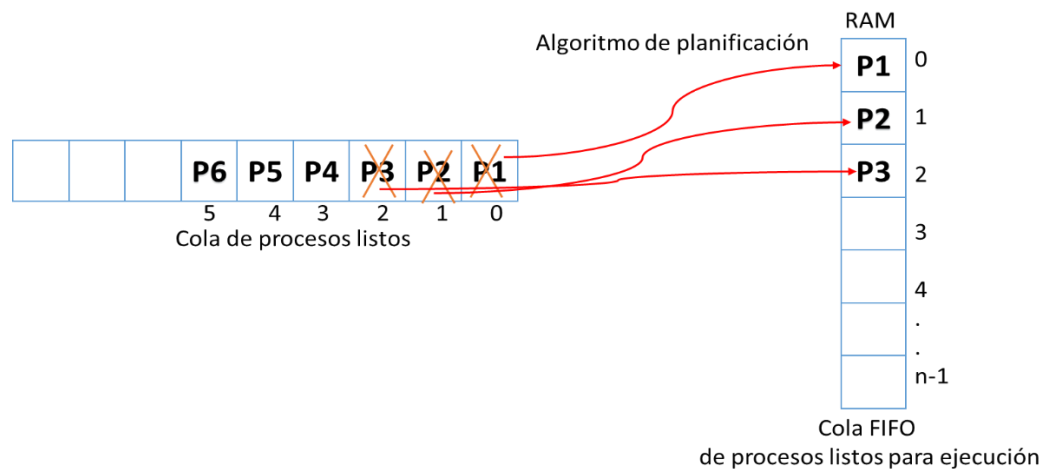




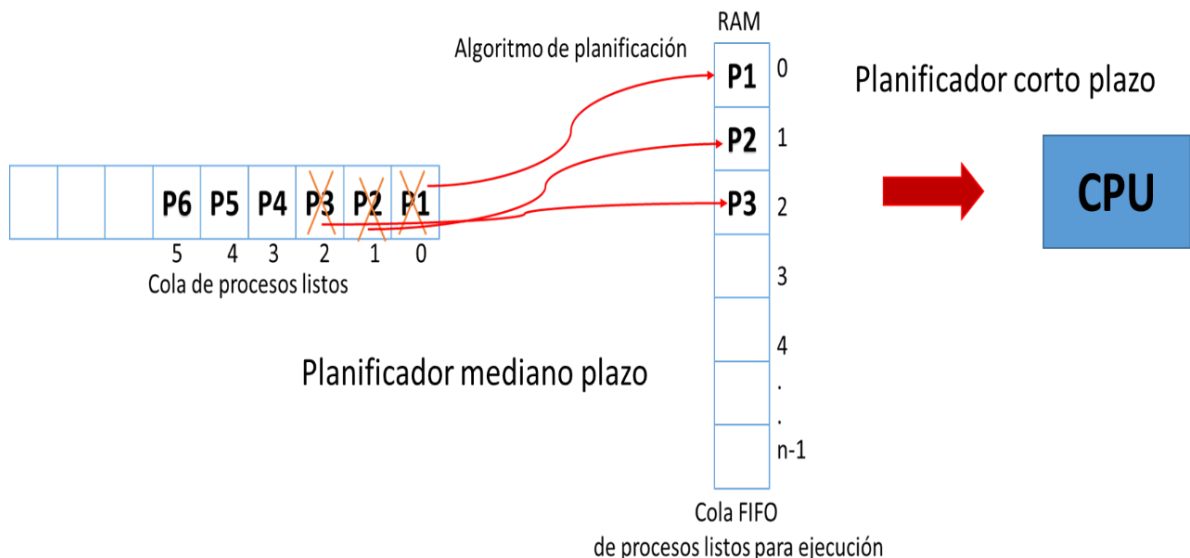
3. Planificación de procesos

➤ Se simulará la política de planificación **Round Robin**

A. Se toma el primer proceso de la **cola de procesos listos** y se sube a la memoria (planificador mediano plazo), registrándolo en la **cola de procesos listos para ejecución**.



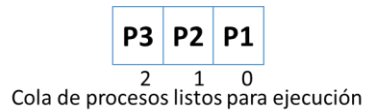
B. De la cola de procesos listos para ejecución se toma el proceso y se le asigna la CPU (planificador corto plazo)



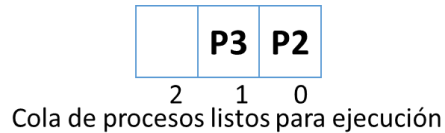
C. Por cada inserción o eliminación de procesos sobre las colas del sistema, estas deben de volver a imprimir su información a pantalla.



Ejemplo: Simulación de la ejecución de los procesos en la CPU:



Se planifica P1:



CPU: Para el ejemplo considere que P1 tiene una ráfaga de CPU de 10 mseg

P1 en ejecución 10 msg
P1 en ejecución 9 msg
P1 en ejecución 8 msg
P1 en ejecución 7 msg
.....

Nota: Cuando el proceso NO ha terminado su ráfaga y se terminó su quantum se debe formar en la **cola de procesos listos**, para competir nuevamente por los recursos y subir nuevamente a la memoria (planificador mediano plazo) y después subir a la cpu (planificador a corto plazo).

4. Resultados finales

Calcular los tiempos promedio de:

- 1) Espera
- 2) Ejecución
- 3) Respuesta

5. Lenguaje de programación JAVA

6. Rúbrica de evaluación

Rúbrica para el Planificador de Procesos
Sistemas Operativos
6º. Semestre de Ingeniería en Computación

Valor	Planificador	Excelente	Bueno	Deficiente (0%)	Puntuación
20%	Planificador mediano plazo Round Robin	✓ Planifica sin errores ✓ Dibuja la cola de procesos listos correctamente (20%)	✓ Planifica sin errores (10%)	✓ Errores en la planificación ✓ Errores o ausencia de la cola de procesos listos	
20%	Planificador de corto plazo Round Robin	✓ Planifica sin errores ✓ Dibuja la cola de procesos listos correctamente (20%)	✓ Planifica sin errores (10%)	✓ Errores en la planificación ✓ Errores o ausencia de la cola de procesos listos	
10%	Simulación de la partición de la memoria	✓ Se presenta a pantalla cada asignación y liberación de memoria. ✓ Se presentan las colas de procesos actualizadas por cada movimiento de la RAM: 1) De espera a la RAM 2) De la RAM A la CPU (10%)	✓ Ausencia de información de las estructuras cola a pantalla (5%)	✓ Ausencia de simulación o bien errores en la simulación de la memoria ✓ Información incompleta	
10%	Tiempos de espera promedio	✓ Funciona correctamente (10%)	✓ Funciona con errores (0%)	✓ No se implemento	
10%	Tiempos de respuesta promedio	✓ Funciona correctamente (10%)	✓ Funciona con errores (0%)	✓ No se implemento	
10%	Tiempos de ejecución promedio	✓ Funciona correctamente (10%)	✓ Funciona con errores (0%)	✓ No se implemento	
10% Obligatorio	Simulación de ejecución en la CPU	✓ Funciona correctamente (10%)	✓ Funciona con errores (10%)	✓ No se implemento	
10 % Obligatorio	Deberá de documentar el proyecto Incluir conclusiones personales	✓ Documentación completa ✓ Conclusiones completas (10%)	✓ Documentación incompleta (0%)	✓ Sin documentación	
TOTAL:					

- 1.- Los rubros marcados como obligatorios, deberán de ser cubiertos al 100% de lo contrario solo obtendrá **50%** de la puntuación total alcanzada en el proyecto.
- 2.- La calificación de los integrantes del equipo será **individual** y dependerá del examen oral sobre su proyecto
- 3.- El proyecto será **cancelado**, colocando una calificación de **cero**, si suceden los siguientes eventos:
 - a) Si se entrega fuera de tiempo.
 - b) Se desarrolla en equipo de 4 personas, de no cumplir este requisito se cancelará.
 - c) Si existen simuladores muy parecidos
 - d) Si no cumple con los requisitos de planificación solicitados
 - e) Si no cumple con las especificaciones del proyecto (sin excepción)