



Alumno:	García González Aarón Antonio
Grupo:	2CM9
Unidad de Aprendizaje:	Sistemas Operativos
Profesor:	Ana Belem Juárez Méndez
Tarea 4:	Investigar un algoritmo de planificación no visto en clase
Fecha:	Lunes 14 de octubre de 19

Primeramente, un recuento muy breve de lo visto en clase:

Modos de decisión; especifica los instantes de tiempo en que se ejecuta la función de selección. Hay dos categorías generales:

Sin expulsión = No apropiativa = No priorizante = Siempre procesa una solicitud programada hasta su terminación

Con expulsión = Apropiativa = Priorizante = Un proceso ejecutando en un determinado momento puede ser interrumpido y pasado al estado de listo por el sistema operativo. La decisión de expulsar puede ser tomada cuando llega un nuevo proceso, cuando llega una interrupción que pasa un proceso de bloqueado a estado de listo, o periódicamente, basándose en las interrupciones del reloj.

Algoritmos vistos en clase:

- Primero en llegar, primero en servirse (**FCFS**): Selecciona el proceso que más tiempo ha estado esperando servicio.
- Turno rotatorio (**RR**): Utiliza rodajas de tiempo para limitar los procesos en ejecución a una pequeña ráfaga de tiempo de ejecución, y rota entre todos los procesos listos.
- Primero el proceso más corto (**SJF**) Selecciona el proceso con el menor tiempo de procesamiento esperado y no expulsa a los procesos.
- Menor tiempo restante (**SRTF**): Selecciona el proceso con el menor tiempo de procesamiento restante esperado. Un proceso puede ser expulsado cuando otro proceso pasa a listo.
- Primero el de mayor tasa de respuesta (**HRN**): Basa la decisión de planificación en una estimación del tiempo de estancia normalizado.
- Retroalimentación (**Colas múltiples**): Establece un conjunto de colas de planificación y sitúa los procesos en las colas basándose en su historia de ejecución y otros criterios.

Planificación por sorteo

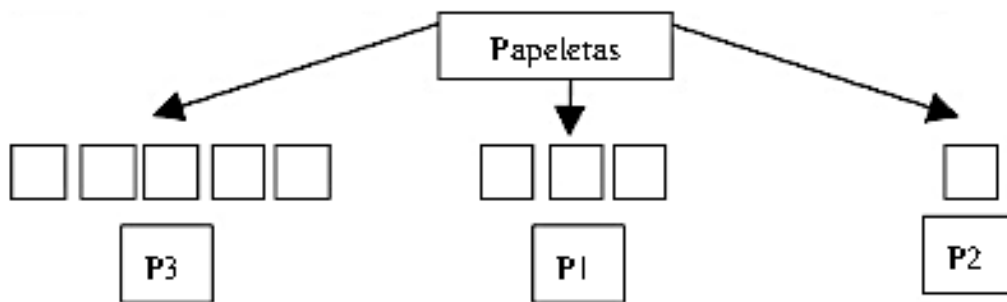
Este algoritmo se conoce como planificación por sorteo (Waldspurger y Weihl, 1994). La idea básica es dar a los procesos boletos de lotería para diversos recursos del sistema, como el tiempo de la CPU. Cada vez que hay que tomar una decisión de planificación, se selecciona un boleto de lotería al azar y el proceso que tiene ese boleto obtiene el recurso. Cuando se aplica a la planificación de la CPU, el sistema podría realizar un sorteo 50 veces por segundo y cada ganador obtendría 20 mseg de tiempo de la CPU como premio. Parafraseando a George Orwell: "Todos los procesos son iguales, pero algunos son más iguales que otros". Los procesos más importantes pueden recibir boletos adicionales, para incrementar su probabilidad de ganar. Si hay 100 boletos repartidos y un proceso tiene 20 de ellos, tendrá una probabilidad de 20 por ciento de ganar cada sorteo. A la larga, recibirá un 20 por ciento del tiempo de la CPU. En contraste a un planificador por prioridad, en donde es muy difícil establecer lo que significa tener una prioridad de 40, aquí la regla es clara: un proceso que contenga una fracción f de los boletos recibirá aproximadamente una fracción f del recurso en cuestión. La planificación por lotería tiene varias propiedades interesantes. Por ejemplo, si aparece un nuevo proceso y recibe algunos boletos, en el siguiente sorteo tendrá la oportunidad de ganar en proporción al número de boletos que tenga. En otras palabras, la planificación por lotería tiene un alto grado de respuesta. Los procesos cooperativos pueden intercambiar boletos si lo desean. Por ejemplo, cuando un proceso cliente envía un mensaje a un proceso servidor y después se bloquea, puede dar todos sus boletos al servidor para incrementar la probabilidad de que éste se ejecute a continuación. Cuando el servidor termina, devuelve los boletos para que el cliente se pueda ejecutar de nuevo. De hecho, en ausencia de clientes, los servidores no necesitan boletos. La planificación por sorteo se puede utilizar para resolver problemas que

son difíciles de manejar con otros métodos. Un ejemplo es un servidor de video en el que varios procesos están enviando flujos continuos de video a sus clientes, pero enviando sus cuadros a distintas velocidades. Suponemos que el proceso necesita cuadros a 10, 20 y 25 cuadros por segundo. Al asignar a estos procesos 10, 20 y 25 boletos, respectivamente, dividirán de manera automática el tiempo de la CPU en la proporción correcta aproximada; es decir: 10 : 20 : 25.

Por ejemplo, Si hay 3 procesos preparados P1, P2 y P3, que cuentan respectivamente con 3, 1 y 5 papeletas, estarán ordenados en la cola como P3, P1 y P2. Cuando hay que elegir un nuevo proceso a planificar, el sistema realiza un sorteo generando un número aleatorio entre 1 y el número total de papeletas repartidas.

Tras el sorteo, encontraremos al ganador si contamos papeletas desde el principio de la cola de preparados. Por ejemplo, si sale ganadora la papeleta séptima, el ganador será el proceso P1.

El proceso ganador tomará control de la CPU hasta que se bloquee voluntariamente o el sistema le expulse porque haya consumido una porción de tiempo preestablecida. El sorteo se repite siempre que haya que elegir un nuevo proceso a planificar, teniendo en cuenta que, en cada sorteo, el número de procesos preparados puede ser distinto y por tanto, el número total de papeletas a considerar también.



Referencias:

- [1]. Planificación por sorteo (lotería). (s.f.). Recuperado 13 octubre, 2019, de <http://algoritmosplanificacion.blogspot.com/2012/08/planificacion-por-sorteo-o-loteria-la.html>
- [2]. 6 Algoritmos de Planificación I. (s.f.). Recuperado 13 octubre, 2019, de <https://lsi.vc.ehu.eus/pablogn/docencia/manuales/SO/TemasSOuJaen/PLANIFICACIONDEPROCESOS/6AlgoritmosdePlanificacionI.htm>
- [3]. Tanenbaum, ANDREW S. (2009). Sistemas operativos modernos. (3a ed.). CDMX, Mexico: PEARSON EDUCACIÓN.