

Investigación 02: Algoritmos de planificación de procesos

SEMINARIO PARA LA SOLUCIÓN DE PROBLEMAS DE SISTEMAS
OPERATIVOS – D06
INGENIERÍA EN COMPUTACIÓN

Algoritmos de planificación de procesos

Cuando un proceso es ejecutable, el sistema operativo debe decidir cuál de ellos, debe ejecutarse en primer término. Esa parte del sistema operativo que debe llevar a cabo esa decisión se llama el planificador y el algoritmo que utiliza se llama algoritmo de planificación.

En los sistemas de procesamiento por lotes, con una entrada de forma de imágenes de tarjeta en una cinta magnética, el algoritmo de planificación era sencillo: sólo había que ejecutar el siguiente trabajo en la cinta, se podía ver como un proceso secuencial. En los sistemas de multiusuario de tiempo compartido, que se combina en un fondo de trabajos procesados en lote, el algoritmo era es complejo. Esa necesidad de poder ordenar los procesos para ganar eficiencia a la hora de tratar con ellos, es lo que llevo a la creación de nuevos algoritmos de planificación, tomando en cuenta los siguientes aspectos.

- Tiempo de espera: El tiempo que un proceso permanece en espera en la cola de ejecución.
- Tiempo de retorno: Tiempo que va desde que se lanza un proceso hasta que finaliza.
- Tiempo de respuesta: el tiempo que un proceso bloqueado tarda en entrar en ejecución.
- Equidad: garantizar que cada proceso obtiene su proporción justa de la CPU. Eficacia: mantener ocupada a la CPU el 100% de tiempo.
- Uso de CPU: Porcentaje de tiempo que la CPU está ocupada.
- Productividad: Número de procesos realizados en una unidad de tiempo.
- Apropiativo: También conocido como expulsivo o expropiativo, este tipo de algoritmo nos permite la expulsión de procesos para ejecutar un nuevo proceso, poniendo en cola al anterior.
- No Apropiativo: Este tipo no nos permite la expulsión, por lo que un proceso nuevo no entrará hasta que termine el anterior.
- Máxima utilización de CPU obtenida con multiprogramación
- La ejecución de procesos consiste de ciclos de ejecución de CPU y esperas en Entradas/Salidas.
- Distribución de ráfagas de CPU

- Round Robin (lapso)

Este algoritmo utiliza un lapso de tiempo llamado quantum. Este es el tiempo máximo que puede estar un proceso en ejecución por vez. Si pasado dicho tiempo el proceso no ha terminado de ejecutarse, el gestor de procesos lo pone en espera, lo vuelve a poner a la cola y entra en ejecución el siguiente proceso. Su principal virtud está en que elimina la posibilidad de que algún proceso no llegue a ejecutarse.

Cada proceso toma una pequeña unidad de tiempo CPU (quantum de tiempo), por lo general de 10-100ms. Después de transcurrido este lapso de tiempo, el proceso es expropiado y ubicado en la cola de listos. Si hay n procesos en la cola de listos y el quantum es q , entonces cada proceso toma $1/n$ de tiempo de CPU en bloques de a lo más q unidades de tiempo a la vez. Ningún proceso espera más que $(n-1) * q$ unidades de tiempo.

- Interrupción por hardware no por software.
- Muy válido para entornos de tiempo compartido.
- Se alterna el uso de la CPU.
- Es muy fácil de implementar.
- Necesita el planificador es mantener una lista de los procesos listos

- SJF (el más corto primero)

Este algoritmo atiende primero a los procesos que necesiten el menor tiempo de ejecución. Una vez un proceso está en ejecución, la CPU no lo deja hasta su finalización. Su principal virtud es que al atender primero a los procesos de menor tiempo de ejecución, el tiempo de espera medio es bastante bajo. Sin embargo, es posible que los procesos más largos suelen tardar muchísimo en entrar en ejecución por lo que sus tiempos de retorno son enormes. Incluso se podría dar el caso de que no entren a ejecución nunca.

Al igual que en el algoritmo FIFO las ráfagas se ejecutan sin interrupción, por tanto, sólo es útil para entornos batch (procesamiento por lotes). Hay que recordar que en los entornos batch se pueden hacer estimaciones del tiempo de ejecución de los procesos, lo que nos permite determinar cuál proceso vamos a ejecutar primero.

Minimiza el tiempo de finalización promedio, como puede verse en el siguiente ejemplo:

- FIFO $F = (24 + 27 + 30) / 3 = 27$ ms.
- SJF $F = (3 + 6 + 30) / 3 = 13$ ms.

Ya que crece de manera exponencial, el tiempo de ejecución del primero es el tiempo de ejecución del segundo más su propio tiempo de ejecución.

No obstante, este algoritmo sólo es óptimo cuando se tienen simultáneamente todas las ráfagas. Contraejemplo, tenemos cinco ráfagas desde A hasta E, con tiempo de ejecución de

2, 4, 1, 1 y 1 respectivamente. Primero se dispone de A y B, puesto que las demás ráfagas no han llegado aún. Con el algoritmo SJF se ejecutan en orden A, B, C, D, y E con un tiempo de finalización promedio de 4.6. Sin embargo, al ejecutarlas en orden B, C, D, E y A se tiene un promedio de finalización de 4.4.

- FIFO (el primero en llegar, el primero en salir)

Los procesos se despachan de acuerdo con su tiempo de llegada a la cola de procesos listos. Cuando un proceso tiene la CPU, se ejecuta hasta terminar. Es justo en el sentido formal, pero algo injusta en cuanto a que los trabajos largos hacen esperar a los cortos y los trabajos sin importancia hacen esperar a los importantes. FIFO ofrece variaciones relativamente pequeñas en los tiempos de respuesta y por lo tanto es más predecible que los otros esquemas.

No es útil en la planificación para los usuarios interactivos porque no puede garantizar buenos tiempos de respuesta por lo que rara vez se usa como esquema principal en los sistemas actuales, pero a menudo está incorporado en otros sistemas. Por ejemplo, muchos esquemas de planificación despachan los procesos de acuerdo con la prioridad, pero los procesos con la misma prioridad se despachan de acuerdo con el esquema FIFO.

Este es un algoritmo que no usa apropiación, y que consiste en atender a los procesos según el orden de llegada a la lista de procesos listos. Cada proceso se ejecuta hasta que termina, o hasta que hace una llamada bloqueante (de E/S).

Las características principales de este algoritmo son las siguientes:

- No es apropiativa.
- Es justa, aunque los procesos largos hacen esperar mucho a los cortos.
- Es una política predecible.
- El tiempo promedio de servicio es muy variable ya que está en función del número de procesos y la duración promedio que tenga.
- Fácil implementación.
- No es válido para entornos interactivos ya que un proceso de mucho cálculo de CPU hace aumentar el tiempo de espera de los demás procesos.

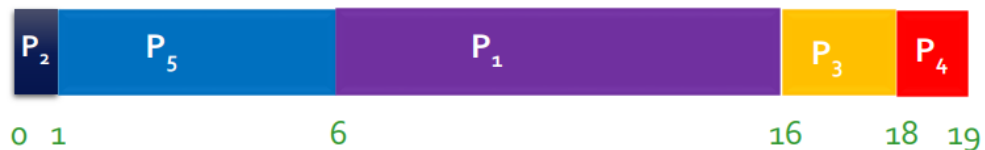
- Prioridades (establece prioridades)

En este algoritmo, a cada proceso se le asocia un número entero de prioridad, la prioridad se caracteriza por el valor numérico, mientras el número sea más pequeño la prioridad es mayor, y mientras más grande la prioridad será menor.

- La CPU es alocada al proceso con prioridad más alta según la convención elegida. Apropiativo.
- No apropiativo
- SJF es un algoritmo de planificación con prioridad.
- Problema de Inanición – los procesos de baja prioridad pueden no llegar a ejecutarse nunca.
- Solución envejecimiento – se incrementa en el tiempo la prioridad de los procesos en espera.

Proceso	Ráfaga	Prioridad
P_1	10	3
P_2	1	1
P_3	2	4
P_4	1	5
P_5	5	2

• La carta de Gantt



- Tiempo de espera promedio= 8.2 msec

Bibliografía

Algoritmos de planificación de procesos. (s/f). Jmoral.es. Recuperado el 9 de septiembre de 2023, de <http://jmoral.es/blog/planificacion-procesos>

*De chosunal, V. T. las E. (2017, septiembre 29). TURNO CIRCULAR(ROUND ROBIN). Blog del curso Sistemas Operativos.
<https://chosunal201720912503.wordpress.com/2017/09/29/turno-circularround-robin/>*

*Jcanbus, P. P. (2014, octubre 7). Pequeña teoría sobre la planificación de procesos. Portfolio Juan José Cánovas Bustamante.
<https://juanjosecanbus.wordpress.com/2014/10/07/pequena-teoria-sobre-la-planificacion-de-procesos-pp/>*

*Studija, R., & Copyright (C) 2007. All rights reserved. (s/f). S.O Unidad-2. Angelfire.com. Recuperado el 9 de septiembre de 2023, de
<https://sistemasoperativos.angelfire.com/html/2.6.1.html>*

*Tema 2.2 Algoritmos de planificación - Plataformas operativas de tecnologías de información - Instituto Consorcio Clavijero. (s/f). Edu.mx. Recuperado el 9 de septiembre de 2023, de
https://cursos.clavijero.edu.mx/cursos/147_poti/modulo2/contenidos/tema2.2.html*

*(S/f-a). Ehu.eus. Recuperado el 9 de septiembre de 2023, de
<https://lsi.vc.ehu.eus/pablogn/docencia/manuales/SO/TemasSOuJaen/PLANIFICACIONDEPROCESOS/6AlgoritmosdePlanificacionI.htm>*

*(S/f-b). Edu.ar. Recuperado el 9 de septiembre de 2023, de
<https://cs.uns.edu.ar/~so/data/apuntes/SO-2020-mod%2007.pdf>*