



Lectura 'Monitors: An Operating System Structuring Concept'

Alvarado Morán Óscar Anuar

Computación Concurrente

IIMAS, 2020-1

Universidad Nacional Autónoma de México

La manera de introducir el concepto que se lleva durante la lectura me parece algo peculiar, creo que en lugar de soltar código se podría plantear una analogía o algo parecido, quizás no sea una lectura introductora pero tampoco creo que es un concepto trivial, me parece que pude entender más el asunto debido a que ya habíamos jugado con algo de lo que hablan pero en forma de hilos y procesos.

Como lo dice el título, la lectura habla sobre 'Monitores', si es que así se puede traducir. El nombre tiene sentido para mí no en el aspecto de una pantalla, sino de algo que monitorea otras cosas, y según yo, que es de lo que va el texto, de administrar procedimientos en un sistema operativo. Para la segunda parte de la lectura, y el final de la primera, nos habla de semáforos y su forma de ser implementados mediante un monitor y la manera de implementar un monitor mediante semáforos. Me parece más profundamente explorada la segunda idea, ya que plantea el proceso de administración de procedimientos mediante semáforos hasta con código. Sin embargo, al final de esta parte comentan que habrá significantes avances en la eficiencia de dicha administración si se opta por no usar semáforos y recurrir a la electrónica directamente (compuertas lógicas con transistores, me imagino) o a los niveles más bajos en cuanto a software.

Para la tercera parte de la lectura retoma el concepto que se vió en la introducción sobre la analogía entre un monitor y la representación de datos. Me lo imagino como que el monitor tiene varios -datos- definidos en sí mismo asociados a variables, funciones o cosas así que administrará por sí solo, que se podría ver como una tabla (la representación de datos que me imagino) que tiene coordenadas con valores específicos de datos.

Para la cuarta parte nos da un ejemplo de lo que se ha estado viendo en el texto y un ejercicio que hemos manejado mucho en clase: el problema del productor - consumidor. Nos plantea que tiene que haber un buffer y obviamente el productor y consumidor, con esperas en el envío de señal para desbloqueo (despertar al productor o consumidor) cuando el buíffer está lleno o vacío. Creo que su forma de ponerlo en código no es tan clara dado que la idea es que sea pseudocódigo según entiendo, creo que entiendo lo que está haciendo debido a que ya hemos jugado con ello en clase, pero sin ello sólo estaría

más confundido.

Luego se enfrenta a un problema que ya hemos planteado en clase que no recuerdo su nombre, pero básicamente es que un proceso es creado y no termina, sino que se queda esperando un recurso necesario para continuar. Dicho problema lo ataca especificando que se tiene que hacer un buen manejo de los 'wait' que le das a un proceso.

Ya para la última parte de la lectura nos muestra más ejemplos, empezando por algo que entendí como un multibuffer, es decir, un buffer de buffers, donde cada buffer puede ser, en el caso del productor - consumidor, llenado y vaciado independientemente, accediendo a cada uno de estos sabiendo la dirección, o más bien, la posición del buffer al que se está accediendo de entre todos los buffers; asimismo se plantea el problema de que no deberían poder acceder, por ejemplo, dos productores al mismo tiempo a un buffer sino que deberían comunicarse para poder entrar a distintos; también nos presenta el problema de que puede ser que el consumidor consuma más rápido de lo que produce el productor, lo que se debe tratar viendo su comportamiento y administrar las esperas entre que uno está dormido o el otro lo está y compensar el sobreconsumo. Luego menciona un problema de administración en un disco duro y su cabezal o algo sí tengo entendido. Para esto se tiene que administrar bien la forma en que el cabezal se mueve para escribir o alterar cosas en el disco duro, pero varias tareas lo quieren hacer al mismo tiempo, por lo que habrá que ver la frecuencia con que se quiere hacer que el cabezal se mueva dependiendo cuánto se quiere mover. Y por último nos da un ejemplo de naves espaciales y su monitoreo desde una base de control. Creo que pudo haber empezado con los ejemplos sin código para adentrarte bien a la idea y después ir desarrollando más formalmente los conceptos y metiendo código, pero bueno, quien soy yo para juzgar.

Ya para el final del documento cuando se presentan las conclusiones me parece bastante cómoda la forma en que enumera las buenas prácticas al querer administrar los procedimientos de un sistema operativo, que en realidad se escalan a cosas simples como el los ejemplos que revisamos. Se podría tomar como una guía paso a paso para crear un buen administrador, o al menos para saber los problemas que te vas a enfrentar ya que alguien ya los ha enfrentado.