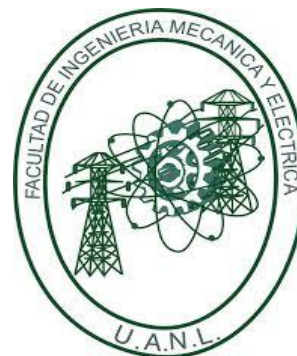




Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica



SISTEMAS OPERATIVOS ACTIVIDAD FUNDAMENTAL 2

Instructor: M.C NORMA EDITH MARIN MARTINEZ

Número y nombre de la Act: (2) Multitarea y control de concurrencia.

Matrícula	Estudiante	Programa Educativo	Semestre	Equipo	Grupo	Hora	Día
2077856	Christopher Angel Santiago Torres	IAS	5to	3	004	N4	M

Día 03 del mes Septiembre del año 2022, San Nicolás de los Garza, Nuevo León.

Índice:

Contenido

Índice:	2
Definición de concurrencia:	3
Tipos de concurrencia:	4
Modelos de programación concurrente:	4
Ventajas de la ejecución de concurrencia:	4
Sistemas operativos de red:	5
¿Por qué es importante la administración de memoria en estos dispositivos?	5
¿Qué debe hacer el sistema operativo para llevar un control de los espacios disponibles y ocupados en la memoria?	6
¿Qué problemas son los más frecuentes en la admon. De memoria?	6
Link del video:	6
Conclusión.....	6
Referencias bibliográficas:	7

Definición de concurrencia:

Es una propiedad de los sistemas en la cual los procesos de un cómputo se hacen simultáneamente, y pueden interactuar entre ellos, es decir son procesados al mismo tiempo, de manera que, para ejecutar uno de ellos, no hace falta que se haya ejecutado otro.

En otras palabras la concurrencia es simplemente la ejecución de varias tareas al mismo tiempo. Ahora esto de ejecutarse “al mismo tiempo” es relativo de acuerdo a la arquitectura del ordenador, -¿espera que? De seguro preguntaras- esto es debido a que de acuerdo a la arquitectura básica, el ordenador solo puede ejecutar un programa o tarea a la vez, y no más, si nosotros persigamos la ilusión de que varios programas se ejecutan al mismo tiempo (Multiprogramación) es debido a que el sistema operativo intercala las tareas que se ejecutan en el procesador a una velocidad sumamente rápida.

Pongamos un ejemplo: digamos que estoy reproduciendo una canción al mismo tiempo que estoy escribiendo esta entrada, y a su vez estoy tengo un video abierto en youtube. Lo que en realidad sucede es que mi computadora hace esto:

- Reproduce la canción durante 1 nanosegundo y se detiene
- Escribe una letra durante 1 nanosegundo y se detiene

- Reproduce una parte del video durante 1 nanosegundo y se detiene
- Repite estas instrucciones otra vez miles de veces por segundo.

Tipos de concurrencia:

Proceso independiente: Es aquel que ejecuta sin requerir la ayuda o cooperación de otros procesos. Un claro ejemplo de procesos independientes son los diferentes shells que se ejecutan de forma simultánea en un sistema.

Procesos son cooperantes: Son aquellos que están diseñados para trabajar conjuntamente en alguna actividad, para lo que deben ser capaces de comunicarse e interactuar entre ellos. En ambos tipos de procesos (independientes y cooperantes), puede producirse una serie de interacciones entre ellos y pueden ser de dos tipos:

- Interacciones motivadas porque los procesos comparten o compiten por el acceso a recursos físicos o lógicos. Por ejemplo, dos procesos independientes compiten por el acceso a disco o para modificar una base de datos.
- Interacción motivada porque los procesos se comunican y sincronizan entre sí para alcanzar un objetivo común, Por ejemplo, un compilador que tiene varios procesos que trabajan conjuntamente para obtener un solo archivo de salida.

Modelos de programación concurrente:

Varias aplicaciones: La multiprogramación se creó para permitir que el tiempo de procesador de la máquina fuese compartido dinámicamente entre varios trabajos o aplicaciones activas.

Aplicaciones estructuradas: Como ampliación de los principios del diseño modular y la programación estructurada, algunas aplicaciones pueden implementarse eficazmente como un conjunto de procesos concurrentes.

Estructura del sistema operativo: Las mismas ventajas de estructuración son aplicables a los programadores de sistemas y se ha comprobado que algunos sistemas operativos están implementados como un conjunto de procesos.

Ventajas de la ejecución de concurrencia:

- Proporciona el modelo más simple y natural de concebir muchas aplicaciones.

- Facilita el diseño orientado a objeto de las aplicaciones, ya que los objetos reales son concurrentes.
- Hace posible compartir recursos y subsistema complejos.
- En sistemas monoprocesador permite optimizar el uso de los recursos.
- Facilita la programación de tiempo real, ya que se pueden concebir como
- Procesos cuya ejecución se planifican de acuerdo con la urgencia.
- Permite reducir los tiempos de ejecución sobre plataformas multiprocesadoras.
- Facilita la realización de programas fiables por despliegue dinámico de los procesos en los procesadores.

Sistemas operativos de red:

Software que controla y hace posible la interconexión entre dos o más ordenadores en forma de red. Esto permite que los usuarios interconectados puedan compartir información del software (como archivos y programas) o bien recursos de hardware.

Cada programa de tipo sistema operativo, incluyendo los sistemas operativos de red, gestionan las interacciones entre las memorias, CPU, etc. Y los dispositivos de hardware. También regulan las solicitudes de servicios y el tiempo de uso de cada aplicación de las computadoras conectadas dentro de la red.

¿Por qué es importante la administración de memoria en estos dispositivos?

- Para optimizar el espacio y poder cargar o intercambiar los programas que van hacer ejecutados del disco duro a la memoria principal.
- El administrador de memoria se encarga de llevar un registro de las partes de la memoria que están en uso y de las que no. Si detecta que hay una parte que ya no está en uso, la libera para poder asignarla a los procesos que la necesiten.

- El administrador de memoria proporciona protección y uso compartido, es decir, facilitar un espacio de memoria para cada proceso y controlar que ninguno de ellos trabaje en zonas de memoria que no le han sido asignados.
- Administrar el intercambio entre la memoria principal y el disco en los casos en los que la memoria principal no le pueda dar capacidad a todos los procesos que tienen necesidad de ella.

¿Qué debe hacer el sistema operativo para llevar un control de los espacios disponibles y ocupados en la memoria?

El SO gestiona la Unidad de Administración de Memoria (MMU), la cual es un dispositivo hardware que transforma las direcciones. La función de la MMU es seguir la pista de las partes de la memoria que están en uso y libre, con el fin de poder asignar memoria a los procesos cuando la necesiten y recuperar esa memoria cuando dejen de necesitarla, así como gestionar el intercambio entre memoria principal y el disco cuando la memoria principal resulte demasiado pequeña para contener a todos los procesos.

¿Qué problemas son los más frecuentes en la admon. De memoria?

1. La reubicación, que se presenta porque la memoria es compartida, y el programador no sabe qué área se asigna, el SO asigna un área de memoria libre.
2. PROTECCIÓN, El SO debe garantizar protección de la memoria, para evitar accesos indebidos.
3. Compartición, el sistema de gestión de memoria debe conceder correctamente a los procesos el bloque que comparten.

Link del video:

<https://youtu.be/2M4leM6EZs8>

Conclusión:

Aprendí que la concurrencia se conforma dos actividades, y para que puedan ser concurrentes, es primordial y necesario que tengan relación entre sí, como puede ser la cooperación en un trabajo determinado o el uso de información compartida y los procesos concurrentes pueden funcionar en forma totalmente independiente

unos de otros, o pueden ser asíncronos pero requieren trabajar en cooperación y tienen grandes ventajas como simplificar el código de forma más simple y claro para nosotros, conocer los tipos de la concurrencia no es práctico por qué nos permite conocer los puntos de la concurrencia física y lógica.

Referencias bibliográficas:

U. (2018, 16 diciembre). Concurrencia y Secuencialidad (Sistemas Operativos). Studentplace. Recuperado 3 de septiembre de 2022, de

<https://studentplace98.blogspot.com/2018/09/concurrencia-y-secuencialidad-sistemas.html?m=1>

Vazquez, J. A. (2001). Ortogonalidad de la concurrencia en modelos de concurrencia orientados a objeto (Doctoral dissertation, Universidade de Vigo).

Oliva Hernández, A. (2020). Transformación de diagramas de comportamiento a modelos de concurrencia (Bachelor's thesis).

Drake, J. M., & Aldea, M. (2001). Programación concurrente. CTR-Computadores y Tiempo Real..[En línea][Citado el: 8 de junio de 2016.] http://www.ctr.unican.es/asignaturas/procodis_3_ii/doc/procodis_1_01. Pdf.

C. (2020, 6 marzo). Sistemas Operativos de Red. ConceptoABC. Recuperado 3 de septiembre de 2022, de <https://conceptoabc.com/sistemas-operativos-de-red/>

Gestión de Memoria - Sistemas Operativos I. (2019, 13 octubre). Sistemas operativos I. Recuperado 3 de septiembre de 2022, de <https://sites.google.com/site/sisoper1/home/gestion-de-memoria#:~:text=%C2%BFPor%20qu%C3%A9%20se%20necesita%20la,y%20de%20las%20que%20no>.