



UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA



UNIDAD DE APRENDIZAJE: SISTEMAS OPERATIVOS

SEMESTRE: AGOSTO-DICIEMBRE 2023




ACTIVIDAD FUNDAMENTAL 2:

MULTITAREA Y CONTROL DE CONCURRENCIA

CATEDRÁTICO: DRA. NORMA EDITH MARIN MARTINEZ

HORA: N4-N6 GRUPO: 004

EQUIPO 4

MATRÍCULA	NOMBRE	CARRERA	
2010184	Andrea Ximena Rivera Aceves	ITS	
2001403	Ricardo Rodríguez Lozano	ITS	
2082321	Jesús Salvador Guzmán Hernández	IAS	

FECHA DE ENTREGA: 27 DE AGOSTO DEL 2023

Índice

Link del video	4
Investigación.....	4
Definición de concurrencia.....	4
Tipos de concurrencia	4
Modelos de programación concurrente y sus características.....	5
Ventajas de la ejecución de la concurrencia	6
Tipos de procesos concurrentes	6
Tipos de interacciones entre los procesos de la concurrencia.....	7
Cuadro sinóptico	9
Conclusión Grupal	10
Conclusiones individuales.....	11
Referencias Bibliográficas	12

Introducción

La multitarea y el control de concurrencia son dos conceptos fundamentales en el ámbito de los sistemas operativos, que juegan un papel crucial en la gestión eficiente de los recursos de una computadora y en la optimización del rendimiento de las aplicaciones. Estos dos conceptos se entrelazan para permitir que múltiples tareas se ejecuten de manera simultánea en un entorno computacional, brindando a los usuarios una experiencia fluida y eficiente.

La multitarea se refiere a la capacidad de un sistema operativo para ejecutar varias tareas o procesos de manera concurrente, lo que significa que múltiples aplicaciones pueden estar activas al mismo tiempo, mientras que la concurrencia, se encarga de coordinar estas tareas de manera que se eviten problemas como bloqueos o condiciones de carrera, que podrían llevar a un mal funcionamiento del sistema.

En esta actividad, exploraremos más a fondo estos dos conceptos en sistemas operativos, comprendiendo cómo se implementan, qué desafíos plantean y cómo contribuyen a la eficiencia de nuestros dispositivos.

Link del video

<https://youtu.be/YVfGSu4olz8>

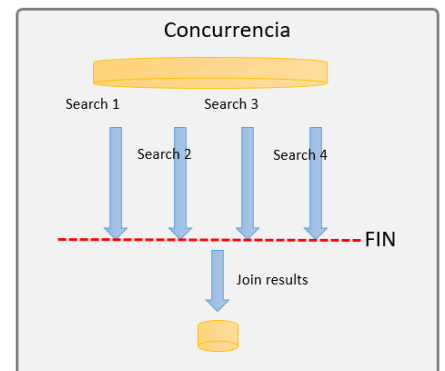
Investigación

Definición de concurrencia

La concurrencia es la capacidad de desglosar un programa en partes que pueden ejecutarse independientemente una de la otra. Esto significa que las tareas se pueden ejecutar sin orden y el resultado seguirá siendo el mismo que si se ejecutaran en el orden original. La concurrencia es la capacidad de un algoritmo o programa para ejecutar más de una tarea a la vez. Es un concepto similar al procesamiento paralelo, pero con la diferencia de que muchos trabajos independientes hagan diferentes cosas a la vez en lugar de ejecutar el mismo trabajo.

La concurrencia hace que el procesador ejecute varias tareas simultáneamente durante un determinado periodo de tiempo. Estos procesos no dependen unos de otros. La concurrencia facilita la transición rápida entre diferentes aplicaciones en un ordenador, por ejemplo; parece que diferentes procesos se están ejecutando exactamente al mismo tiempo. Pero, en cambio, la CPU está cambiando rápidamente entre hilos, que son pequeños segmentos de procesos informáticos. Técnicamente, una CPU sólo puede ejecutar un hilo a la vez, pero la concurrencia permite que este pueda alternar entre las tareas para que parezca que se ejecutan al mismo tiempo. A esto se le llama multihilo.

Un programa concurrente se formula utilizando bien un lenguaje concurrente (como Ada), o definiendo independientemente los procesos que constituyen el programa mediante un lenguaje secuencial (como el lenguaje C), y haciendo uso de la capacidad de multiprocesador que proporciona un sistema operativo multiprocesador (como UNIX) o un software de comunicación (como MMS).



Tipos de concurrencia

- **Concurrencia Física:** Existe más de un procesador y varias unidades (hilos) de un mismo programa se ejecutan realmente de forma simultánea.
- **Concurrencia Lógica:** Asumir la existencia de varios procesadores, aunque no existan físicamente. El implementador de tareas del lenguaje se encargará de "mapear" la concurrencia lógica sobre el hardware realmente disponible. La concurrencia lógica es más general, pues el diseño del programa no está condicionado por los recursos de computación disponibles.
- **Concurrencia a nivel de Datos:** La concurrencia a nivel de datos se refiere a la capacidad de varios procesos o hilos para acceder y modificar datos

compartidos de manera segura. Esto generalmente implica el uso de mecanismos de sincronización, como semáforos, mutex o barreras, para evitar problemas de concurrencia como las condiciones de carrera.

- **Concurrencia a nivel de E/S (Entrada/Salida):** En sistemas operativos modernos, muchas operaciones de E/S (lectura/escritura de archivos, comunicación de red, etc.)
- **Concurrencia a nivel de Red:** En sistemas operativos de red, la concurrencia se utiliza para manejar múltiples conexiones de red de manera simultánea.

Modelos de programación concurrente y sus características

La programación concurrente es un paradigma de programación que se utiliza para diseñar sistemas que pueden ejecutar múltiples tareas de forma simultánea. Esto permite aprovechar la capacidad de las computadoras modernas con múltiples núcleos de CPU para realizar tareas de manera más eficiente. Aquí tienes algunos modelos de programación concurrente comunes y sus características:

1. Threads (Hilos):

- **Características:** Los hilos son la unidad más pequeña de ejecución dentro de un proceso. Comparten el mismo espacio de memoria y recursos, lo que facilita la comunicación y la sincronización.

2. Procesos:

- **Características:** Los procesos son programas independientes que se ejecutan en su propio espacio de memoria. Son más aislados y seguros que los hilos, pero la comunicación entre procesos suele ser más costosa.

3. Programación Orientada a Mensajes:

- **Características:** Los procesos o hilos se comunican enviándose mensajes. No comparten memoria directamente, lo que puede aumentar la robustez y la escalabilidad.

4. Programación Paralela Explícita:

- **Características:** Se utiliza cuando se necesita un control detallado sobre la ejecución paralela. El programador especifica explícitamente qué partes del código deben ejecutarse en paralelo.

La elección del modelo de programación concurrente depende de las características y requisitos específicos de tu aplicación. Es importante considerar la complejidad, el rendimiento y la facilidad de desarrollo al seleccionar un modelo. Además, es común combinar varios de estos modelos en una única aplicación para aprovechar sus ventajas en diferentes partes del sistema.



Ventajas de la ejecución de la concurrencia

La concurrencia en sistemas operativos ofrece una serie de ventajas que son fundamentales para la eficiencia, la capacidad de respuesta y la capacidad de manejar múltiples tareas. Aquí tienes algunas de las principales ventajas de la concurrencia en sistemas operativos:

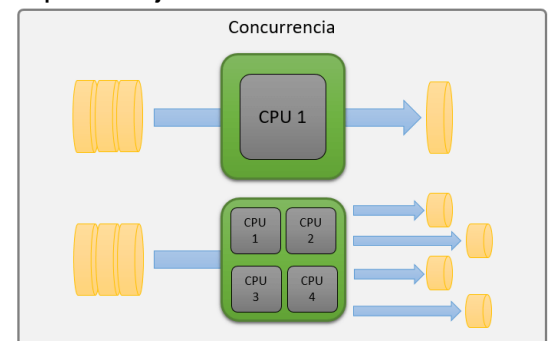
- **Mejora del rendimiento:** Una de las ventajas más significativas de la concurrencia es la mejora del rendimiento del sistema.
- **Tiempo compartido:** La concurrencia permite la implementación de sistemas de tiempo compartido, lo que significa que múltiples usuarios o aplicaciones pueden utilizar el sistema de manera aparentemente simultánea.
- **Multitarea:** La capacidad de ejecutar múltiples tareas simultáneamente es fundamental en sistemas operativos modernos.
- **Paralelismo:** En sistemas con múltiples núcleos de CPU, la concurrencia permite aprovechar al máximo el paralelismo, lo que mejora aún más el rendimiento. Los procesos pueden distribuirse entre los núcleos de la CPU, lo que acelera la ejecución de tareas intensivas en CPU.
- **Mayor capacidad de respuesta:** La concurrencia asegura que un proceso no bloquee completamente el sistema.
- **Facilita la programación modular:** La concurrencia facilita la creación de sistemas modulares y componentes independientes.

La concurrencia en sistemas operativos es esencial para mejorar el rendimiento, la capacidad de respuesta y la capacidad de manejar múltiples tareas simultáneamente. Estas ventajas son cruciales en entornos de computación modernos, donde la multitarea y el paralelismo son fundamentales.

Tipos de procesos concurrentes

Los procesos concurrentes en sistemas operativos pueden clasificarse en varios tipos según su naturaleza y comportamiento. La elección del tipo de proceso concurrente depende de los requisitos y objetivos específicos de la aplicación o sistema en cuestión. Cada tipo de proceso tiene sus propias características y desafíos de programación y gestión.

- **Procesos Independientes:** Estos son procesos que se ejecutan de manera completamente independiente uno del otro, sin compartir datos ni recursos. Cada proceso realiza su propia tarea sin afectar a otros procesos.
- **Procesos Cooperativos:** Los procesos cooperativos trabajan juntos y comparten datos y recursos para lograr un objetivo común. Estos procesos suelen comunicarse entre sí y colaborar para realizar tareas más grandes.



- **Procesos Paralelos:** Los procesos paralelos se ejecutan al mismo tiempo, ya sea en múltiples núcleos de CPU o en múltiples máquinas.
- **Procesos Concurrentes en Lote (Batch):** Estos procesos se utilizan comúnmente en entornos de procesamiento por lotes.
- **Procesos de Usuario:** Los procesos de usuario son procesos que se inician y gestionan por parte de los usuarios o aplicaciones de usuario.
- **Procesos del Sistema:** Estos son procesos que son esenciales para el funcionamiento del sistema operativo en sí.
- **Procesos en Tiempo Real:** Los procesos en tiempo real son procesos que deben completarse dentro de plazos estrictos.
- **Procesos de Comunicación:** Estos procesos se centran en la comunicación entre procesos.

Tipos de interacciones entre los procesos de la concurrencia

En los sistemas operativos, la concurrencia se refiere a la ejecución simultánea de múltiples procesos o hilos. Durante la ejecución concurrente, los procesos pueden interactuar entre sí de varias maneras. Aquí hay algunos tipos de interacciones comunes entre los procesos de concurrencia en sistemas operativos:

1. Comunicación entre procesos (IPC - Inter-Process Communication):

- **Comunicación por tuberías (Pipes):** Los procesos pueden enviar datos de un extremo de una tubería al otro. Esto se utiliza a menudo para la comunicación entre un proceso padre e hijo.
- **Comunicación por sockets:** Los procesos pueden comunicarse a través de sockets de red, ya sea en la misma máquina o a través de una red.
- **Memoria compartida:** Los procesos pueden compartir una región de memoria, lo que les permite intercambiar datos directamente.

2. Sincronización de procesos:

- **Semáforos:** Los semáforos son una forma de controlar el acceso a recursos compartidos. Los procesos pueden bloquearse en semáforos hasta que se cumplan ciertas condiciones.
- **Mutex (Mutexes):** Los mutexes son objetos que permiten a los procesos bloquear el acceso exclusivo a recursos compartidos.
- **Variables de condición:** Se utilizan en combinación con mutexes para permitir que los procesos esperen a que ocurran ciertas condiciones antes de continuar.

3. Planificación de procesos:

- **Prioridades de proceso:** Los sistemas operativos pueden asignar prioridades a los procesos, lo que afecta a la secuencia en la que se ejecutan.
- **Planificación de round-robin:** Los procesos se ejecutan en un orden cíclico, lo que garantiza un reparto equitativo del tiempo de CPU.
- **Planificación basada en eventos:** Los procesos pueden estar esperando eventos específicos antes de continuar su ejecución.

4. Exclusión mutua:

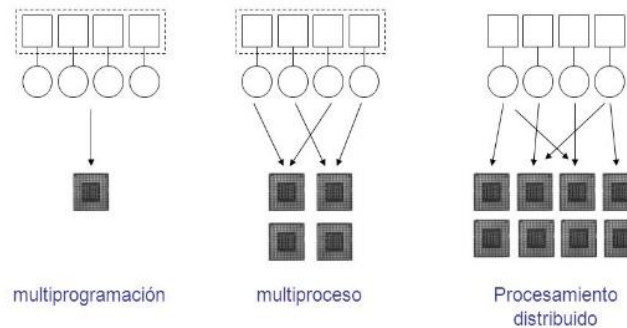
- **Exclusión mutua de hardware:** Utiliza instrucciones de máquina especiales para garantizar que solo un proceso pueda acceder a un recurso compartido a la vez.
- **Exclusión mutua de software:** Los procesos utilizan técnicas como mutexes o semáforos para evitar el acceso concurrente a recursos compartidos.

5. Deadlocks:

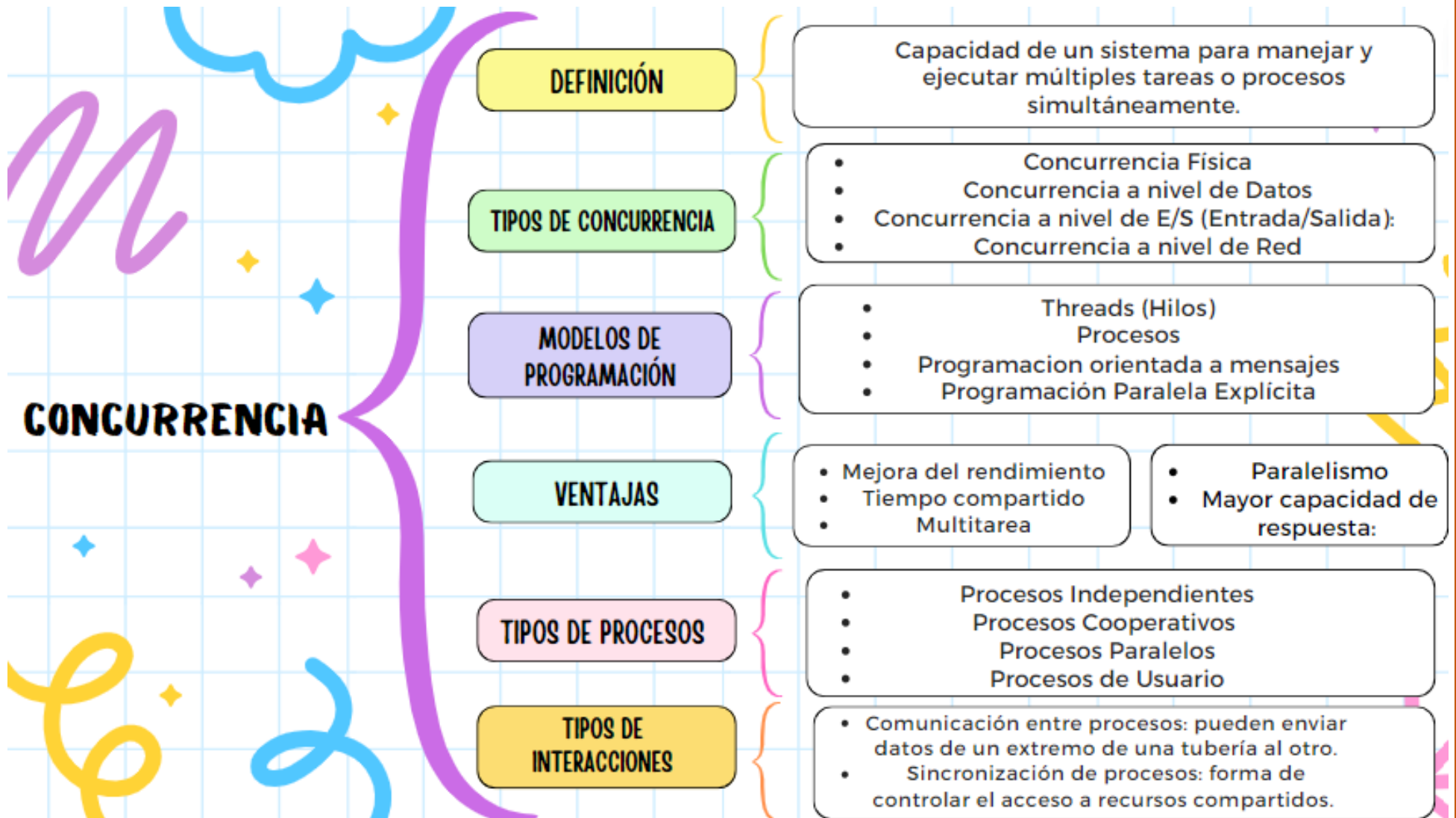
- Los deadlocks son situaciones en las que varios procesos están bloqueados indefinidamente porque cada uno está esperando que otro libere un recurso que necesita. La prevención y detección de deadlocks es crucial en la concurrencia.

Estas son algunas de las interacciones clave que pueden ocurrir entre los procesos en un sistema operativo concurrente. La gestión adecuada de estas interacciones es fundamental para garantizar un funcionamiento estable y eficiente del sistema.

Concurrencia



Cuadro sinóptico



Elaborado en CANVA: https://www.canva.com/design/DAFsmhtwU0/-7qjXKGXbiLVpkXV2QxPZA/edit?utm_content=DAFsmhtwU0&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Conclusión Grupal

En conclusión, la concurrencia en sistemas operativos es un elemento fundamental para garantizar la eficiencia y la estabilidad en entornos computacionales modernos. Permite que múltiples tareas se ejecuten de manera aparentemente simultánea, brindando a los usuarios la capacidad de realizar diversas actividades sin problemas, aunque tiene diversas complicaciones como el coordinar el acceso a recursos compartidos, como la CPU y la memoria, de manera que se eviten conflictos y problemas como bloqueos. Sin embargo, los sistemas operativos tienen diversas estrategias que permiten una distribución de los recursos y previenen problemas.

La gestión efectiva de la concurrencia es muy importante ya que brinda a los usuarios una experiencia informática fluida y eficiente. A medida que la tecnología continúa avanzando y las demandas de rendimiento aumentan, la concurrencia seguirá siendo un área crítica de desarrollo en sistemas operativos, asegurando que los sistemas puedan hacer frente a la creciente complejidad de las aplicaciones y las tareas que se ejecutan en ellos.

Conclusiones individuales

Andrea Ximena Rivera Aceves

En conclusión, a multitarea y la concurrencia en sistemas operativos son esenciales para permitir que varias tareas se ejecuten al mismo tiempo sin problemas. La multitarea permite a los usuarios realizar diversas actividades simultáneamente, mientras que el control de concurrencia garantiza que los recursos compartidos se utilicen de manera ordenada para evitar conflictos. Estos conceptos son vitales en la informática moderna, brindando eficiencia y estabilidad en entornos cada vez más demandantes. Su gestión efectiva permite a los sistemas operativos satisfacer las crecientes necesidades de rendimiento de las aplicaciones y proporcionar una experiencia de usuario fluida.

Ricardo Rodríguez Lozano

La concurrencia es un fenómeno muy interesante e importante, lo podemos ver no solo en la programación sino, también en las empresas, donde podemos separar a las divisiones de la empresa y estas pueden trabajar con autonomía sin depender totalmente de otra división; así pasa con los programas concurrentes, ya que estos los puedes desglosar y ejecutarlos por partes, y estos deberían de poder funcionar y realizar sus tareas independientemente.

Jesús Salvador Guzmán Hernández

En la programación concurrente no es más que la forma en la cual podemos resolver ciertas problemáticas de forma concurrente, es decir, ejecutando múltiples tareas a la misma vez y no de forma secuencial.

En un programa concurrente las tareas pueden continuar sin la necesidad que otras comiencen o finalicen. Si bien es cierto que la programación concurrente acarrea ciertos problemas, principalmente al momento de compartir información entre tareas, también es cierto que, si se implementa de forma correcta, podremos, en casos puntuales, mejorar significativamente el performance de nuestras aplicaciones.

Referencias Bibliográficas

- *Tema 2.5.1 Multitarea, multiprogramación y multiproceso - Tema 2.5 Programación concurrente - Sistemas operativos - Instituto Consorcio Clavijero.* (s. f.). https://cursos.clavijero.edu.mx/cursos/182_so/modulo2/contenidos/tema2.5.1.html?opc=4
- Unknown. (2018, 15 diciembre). Concurrencia y Secuencialidad (Sistemas Operativos). *StudentPlace*. <https://studentplace98.blogspot.com/2018/09/concurrencia-y-secuencialidad-sistemas.html#:~:text=Concurrencia%3A%20es%20una%20propiedad%20de,que%20se%20haya%20ejecutado%20otro.>
- *Tema 4.2 Concurrencia - Módulo 4 - Sistemas en Tiempo Real. - Instituto Consorcio Clavijero.* (s. f.). https://cursos.clavijero.edu.mx/cursos/202_str/modulo4/contenidos/tema4.2.html
- Velázquez, D. (2022, 22 enero). Concurrencia de procesos • sistemas operativos • WebProgramación consultoría informática. *WebProgramación Consultoría Informática*. <https://webprogramacion.com/concurrencia-de-procesos/>
- MANUALES TECH. (2023, August 19). ▷ *Concurrencia, definición en informática* » MANUALES TECH. <https://manualestech.com/definicion/concurrencia/>
- ¿Qué es la concurrencia? - definición de techopedia - Desarrollo 2023. (2023). Icy Science. <https://es.theastrologypage.com/concurrency>