

**PERIODO ESCOLAR: ENERO-JUNIO 2024**

**CATEDRÁTICO: NORMA EDITH MARIN MARTINEZ**

<b>ACTIVIDAD FUNDAMENTAL 2</b>
<b>MULTITAREA Y CONTROL DE CONCURRENCIA</b>

FOTO	NOMBRE	MATRICULA	CARRERA
	Diego Tristan Castro Franco	2109462	IAS
	Angel Daniel Martínez Landeros	1969473	IAS
	Guillermo Vladimir Flores Báez	2127967	ITS
	Jorge Isaac De León Pérez	1932783	ITS

GRUPO	EQUIPO	CALIF	HORA
009	3		M1

Cd. Universitaria, San Nicolas de los Garza, Nuevo León, 07 de Febrero del 2024

## INDICE

INTRODUCCION .....	3
ENLACE DE VIDEO .....	4
¿QUE ES LA CONCURRENCIA? .....	4
TIPOS DE CONCURRENCIA .....	4
MODELOS DE PROGRAMA CONCURRENTES .....	4
EJEMPLO DE CONCURRENCIA .....	5
EJEMPLO DE PARALELISMO .....	5
VENTAJAS DE LA EJECUCION DE CONCURRENCIA .....	6
TIPOS DE PROCESOS CONCURRENTES .....	6
¿CUALES SON LOS TIPOS DE INTERACCIONES ENTRE LOS PROCESOS DENTRO DE LA CONCURRENCIA? .....	6
CUADRO SINOPTICO.....	8
BIBLIOGRAFIA.....	9

## INTRODUCCION

Dentro de la elaboración de este trabajo se hablara de la concurrencia y en una sección se vera el enlace de un video que hablara sobre la concurrencia y se describirán las distintas formas de trabajar los procesos de un sistema operativo, como: procesos e hilos, estados, transiciones, control, gestión del multiprocesamiento simétrico, micro núcleos, gestión de hilos y SMP, así mismo se continua con una investigación acerca de la concurrencia, la cual se ve desde distintos puntos, desde su definición hasta que tipos existen.

La concurrencia de procesos se refiere a situaciones en las que dos o mas procesos pueden coincidir en el acceso a un recurso compartido o, dicho de otra forma, que requieran coordinarse en su ejecución.

## ENLACE DE VIDEO

<https://youtu.be/zZ9hr3kBHVk>

### ¿QUE ES LA CONCURRENCIA?

La concurrencia es la tendencia de las cosas a producirse al mismo tiempo en un sistema y es un fenómeno natural, por supuesto. En la realidad, en un momento dado, suceden muchas cosas simultáneamente. Cuando se diseña un software para supervisar y controlar los sistemas del mundo real, hay que enfrentarse a esta concurrencia natural.

En referencia temas de concurrencia en un sistema software, hay dos aspectos importantes en general; poder detectar y responder a los sucesos externos que se producen en un orden aleatorio y garantizar que estos sucesos se responden en un intervalo requerido mínimo.

Los retos de diseñar sistemas concurrentes derivan de las interacciones que producen entre actividades concurrentes. Cuando las actividades concurrentes interactúan, es necesario algún tipo de coordinación.

Un ejemplo caro de la concurrencia en funcionamiento son las actividades paralelas que no interactúan y que tiene asuntos de concurrencia simple. La concurrencia hace más compleja cuando las actividades paralelas interactúan o comparten los mismos recursos.

### 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:** Asume la existencia de varios procesadores, aunque no existan físicamente. El implementador de tareas de lenguaje se encarga de “mapear” la concurrencia lógica sobre el hardware realmente disponible.

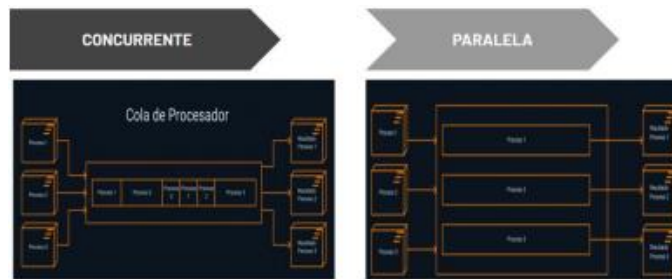
### MODELOS DE PROGRAMA CONCURRENTES

- **Multiprogramación con un único procesador:** el sistema operativo se encarga de repartir el tiempo entre los procesos (planificación expulsiva/no expulsiva).
- **Multiprocesador:** Se combinan en paralelismo real y pseudoparalelismo normalmente más procesos que CPU
- **Sistema distribuido:** varios computadores conectados por red.

La concurrencia y el paralelismo han sido extensamente discutidos, pero al momento de definirlos, varios autores salen con su propia definición y esta variedad no hacen tan fácil la tarea de entender. En términos sencillos, se puede explicar así:

- Un programa es concurrente si puede soportar dos o más acciones en progreso

Un programa es paralelo si puede soportar dos o más acciones ejecutándose simultáneamente.



La palabra clave es un progreso. Un programa es concurrente porque maneja varias tareas al mismo tiempo, define acciones que pueden ser ejecutadas al mismo tiempo.

Y para que un programa sea paralelo, no solo debe ser concurrente, sino que también debe estar diseñado para correr en un medio con hardware paralelo (CPU procesadores multi-core, etc).

Puede verse como que la concurrencia es propiedad de un programa, mientras que el paralelismo es la forma en la que se ejecuta un programa concurrente.

Múltiples procesos pueden ser ejecutados al mismo tiempo. En otras palabras, la programación concurrente se comporta igual que la paralela cuando tenemos un sistema multiprocesador en el cual cada unidad de procesamiento ejecuta un proceso o hilo.

### EJEMPLO DE CONCURRENCIA

Imaginemos una aplicación que descarga música, la cual puede descargar un número determinado de canciones al mismo tiempo, cada canción es independiente de la otra, por lo que la velocidad y el tiempo que tarde en descargarse cada uno no afectará al resto de canciones. Esto se puede ver como un proceso concurrente, ya que cada descarga es un proceso totalmente independiente del resto.



### EJEMPLO DE PARALELISMO

Imaginemos una clásica pagina de viajes, donde nos ayudan a buscar un vuelo mas barato o las mejores promociones, para hacer esto, la pagina debe buscar al momento en cada aerolínea el vuelo mas barato, con menos conexiones, etc. Para esto se puede hacer de dos formas, buscar secuencialmente en cada aerolínea las mejores promociones (muy tardado) o utilizar el paralelismo para buscar al mismo tiempo las mejores promociones en todas las aerolíneas.



## VENTAJAS DE LA EJECUCION 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 subsistemas complejos.
- En sistemas monoprocesador permite optimizar el uso de recursos.
- Facilita la programación en tiempo real, ya que se pueden concebir como procesos cuya ejecución se planifica de acuerdo con la urgencia.
- Permite reducir tiempos de ejecución sobre plataformas multiprocesadoras.
- Facilita la realización de programas fiables por despliegue dinámico de los procesos en los procesadores.

## TIPOS DE PROCESOS CONCURRENTES

### Independientes

- Procesos que ejecuta concurrentemente, pero sin ninguna relación. No necesitan comunicarse.
- No necesitan sincronizarse.
- Ejemplo: Dos interpretes de mandatos de dos usuarios ejecutados en destinos terminales

### Cooperantes

- Procesos que ejecutan concurrentemente con alguna interacción entre ellos. Pueden comunicarse entre si
- Pueden sincronizarse
- Ejemplo: servidor de transacciones organizado en proceso receptor y procesos de tratamiento de peticiones.

## ¿CUALES SON LOS TIPOS DE INTERACCIONES ENTRE LOS PROCESOS DENTRO DE LA CONCURRENCIA?

### Acceso a recursos compartidos

- Procesos que comparten un recurso.
- Proceso que compiten por un recurso
- Ejemplo: servidor de peticiones en las que distintos procesos escriben en un registro de actividad (log).

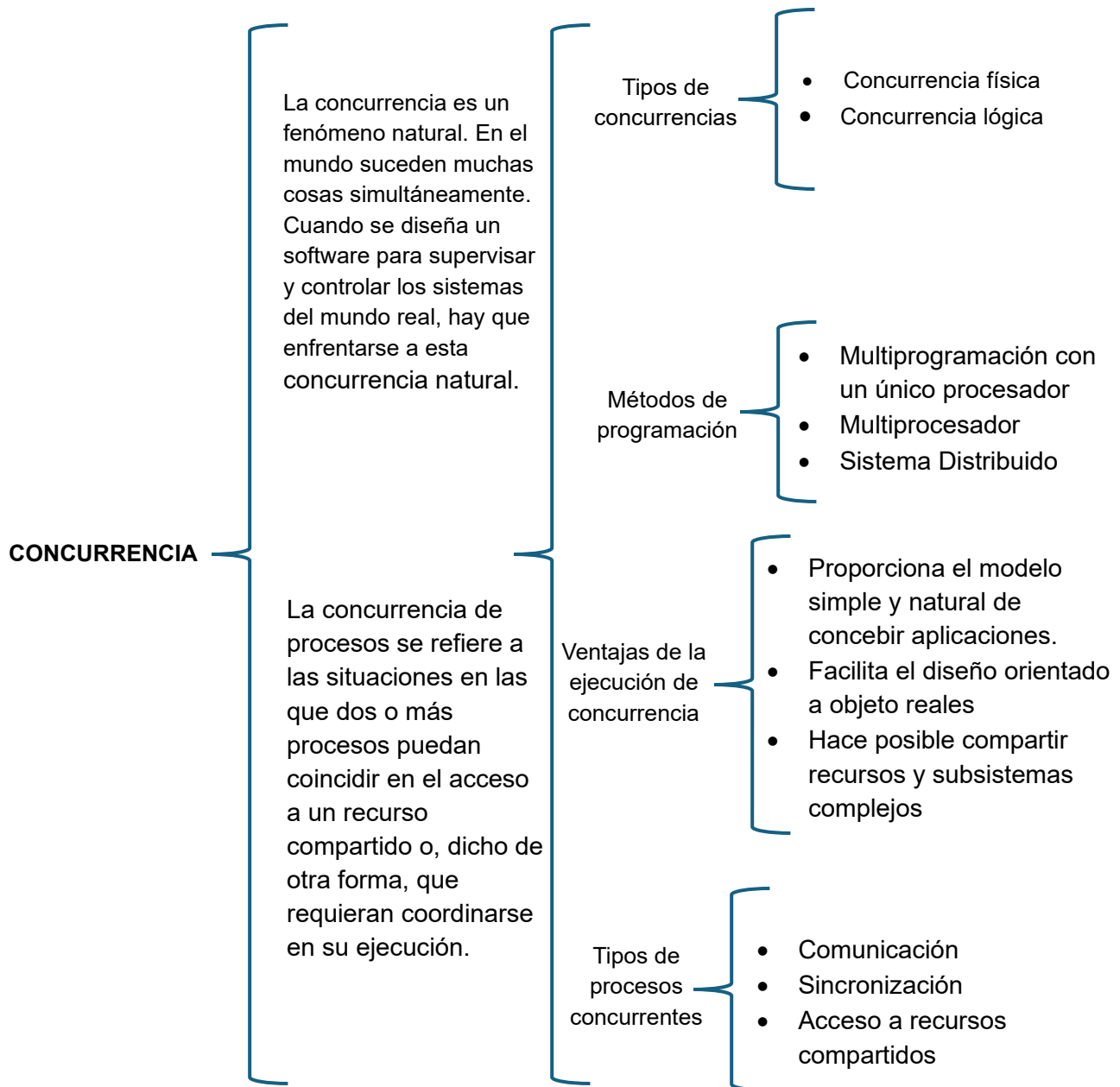
### Comunicación

- Procesos que intercambian información.
- Ejemplo: receptor de peticiones debe pasar información a proceso de tratamiento de petición

### Sincronización

- Un proceso debe esperar a un evento en otro proceso.
- Ejemplo: un proceso de presentación debe esperar a que todos los procesos de cálculo terminen

## CUADRO SINOPTICO





## BIBLIOGRAFIA

[https://cgrw01.cgr.go.cr/rup/RUP.es/SmallProjects/core.base\\_rup/guidances/concepts/c\\_oncurrency\\_EE2E011A.html](https://cgrw01.cgr.go.cr/rup/RUP.es/SmallProjects/core.base_rup/guidances/concepts/c_oncurrency_EE2E011A.html)

[http://ocw.uc3m.es/ingenieriainformatica/sistemas-operativos/material-de-clase-1/mt\\_t3\\_l6.pdf](http://ocw.uc3m.es/ingenieriainformatica/sistemas-operativos/material-de-clase-1/mt_t3_l6.pdf)

[http://informatica.uv.es/iiguia/LP/teoria/apuntes/cuatr1/tema3\\_1\\_concurrencia.pdf](http://informatica.uv.es/iiguia/LP/teoria/apuntes/cuatr1/tema3_1_concurrencia.pdf)

[http://ferestrepoca.github.io/paradigmas-deprogramacion/progconcurrente/concurrente\\_teoría/index.html](http://ferestrepoca.github.io/paradigmas-deprogramacion/progconcurrente/concurrente_teoría/index.html)

[https://www.ctr.unican.es/asignaturas/procodis\\_3\\_ii/doc/procodis\\_1\\_01.pdf](https://www.ctr.unican.es/asignaturas/procodis_3_ii/doc/procodis_1_01.pdf)