

1. La concurrencia en sí es la capacidad de procesar varios procesos al mismo tiempo, por ejemplo si tenemos un procesador de 4 núcleos, mediante la concurrencia se podrían ejecutar 4 procesos diferentes, mientras que el paralelismo es tomar un proceso dividirlo en la cantidad de hilos que tenga el procesador, luego ejecutar cada parte del proceso al mismo tiempo y mediante un proceso final unir todos los resultados, en este caso el proceso se dividirá en 4 partes, se ejecutaría en los 4 núcleos simultáneamente y luego los resultados de cada proceso se van a unir dando por resultado el proceso terminado.

2. Ventajas:

- Se pueden compartir recursos y hacer subsistemas complejos.
- En sistemas de un núcleo se puede optimizar el uso de los recursos.
- Hace la etapa de desarrollo más fácil dado que se pueden crear procesos cuya ejecución se planifica de acuerdo con la prioridad.
- Permite reducir los tiempos de ejecución en plataformas multinúcleo.
- Facilita la creación de programas fiables por despliegue dinámico de procesos en los procesadores.

Problemas:

- Violación de la exclusión mutua: En general la violación a la exclusión mutua es cuando más de un hilo trata de ejecutar la sección crítica del programa y lo logra, obteniendo resultados incorrectos.
  - Deadlock: Ocurre cuando un proceso espera un evento que nunca va a pasar.
  - Aplazamiento indefinido: se da cuando un algoritmo que maneja los recursos no tiene en cuenta el tiempo que lleva esperando el proceso.
  - Injusticia: ocurre cuando el programa no tiene mecanismos/formas para asegurar que se da un progreso uniforme en la ejecución de las tareas concurrentes.
3. En general considero que la concurrencia es indispensable, uno debido a que es aplicable a cualquier sistema, dos que trabajó en aplicaciones web y siempre es necesario agregar concurrencia debido a que las aplicaciones web en la mayoría de sus casos son sistemas que deben reaccionar a múltiples sucesos externos en tiempo real.
4. Los semáforos son estructuras que permiten sincronizar dos o más procesos. En general los semáforos son una estructura compuesta por un contador que inicia en 0 y una cola de procesos que está vacía, en este caso en el semáforo cargaremos procesos y mediante el contador lo sincronizaremos hasta que se quede sin procesos para ejecutar.
5. Una cosa que se podría agregar serían los monitores, que son estructuras que al igual que los semáforos sirven para sincronizar uno o más procesos, pero en este caso no se sincronizan por un contador, sino por una condición, el monitor estaría esperando a que la condición se cumpla, y cuando se cumpla pasaría al siguiente proceso.