

## *Bloque 2. Concurrencia - Introducción*

### *Tarea 4:*

Definir el concepto de **liveness** en concurrencia. Poner algún ejemplo.

Se dice que un programa concurrente disfruta de “vitalidad” cuando responde ágilmente a las solicitudes del usuario, que algo bueno ocurrirá durante la ejecución de los hilos. Esta propiedad de vivacidad nos asegura que algo eventualmente ocurrirá en el sistema.

Esta propiedad es relativa de que cada sentencia que se ejecute conduce en algún modo a un avance constructivo para alcanzar el objetivo funcional del programa. Son en general muy dependientes de la política de planificación que se utilice.

Algunos **ejemplos** de liveness (principio de vitalidad) son:

- **Ausencia de Inanición** – Lo ‘bueno’ es que el proceso progresa.
- **Terminación** – Lo ‘bueno’ es ejecutar la última instrucción.
- **Garantía de Servicio** – Lo ‘bueno’ es que el proceso es atendido.
- **Livelock** – No deben producirse bloqueos activos. Por bloqueos activos se entienden conjuntos de procesos que ejecutan de forma continuada sentencias que no conducen a un progreso constructivo.
- **Starvation (Aplazamiento indefinido)** – consiste en el estado al que puede llegar un programa que aunque potencialmente puede avanzar de forma constructiva, no lo hace como consecuencia de que no se le asigna tiempo de procesador en la política de planificación.