

## Ejercicio 1

a) Si tienes una CPU con 4 núcleos y estás ejecutando 10 procesos, ¿están todos ejecutándose en paralelismo real? Justifica tu respuesta.

No, no puede haber paralelismo real para los 10 procesos ya que al solo disponer de 4 núcleos sólo se puede ejecutar a la vez 4 procesos

b) ¿Y si la CPU tuviese 8 núcleos y estuvieses ejecutando 8 procesos?

En este caso sí que se estarían ejecutando en paralelismo real puesto que hay un núcleo para cada proceso.

## Ejercicio 2

a) Elabora una tabla comparativa entre las características que corresponden a la programación distribuida, paralela o ambas. Por ejemplo:

	Paralela	Distribuida	Ambas
Alta disponibilidad		X	
Ejecuta múltiples tareas al mismo tiempo			X
Uso de múltiples computadoras interconectadas		X	
Comunicación entre procesos			X
Incrementa la velocidad de procesamiento	X		
Utiliza concurrencia			X
Tolerancia a fallos		X	
Recursos compartidos en memoria	X		

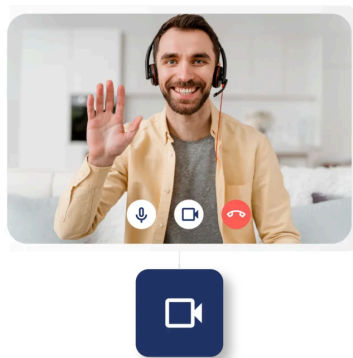
b) VisaNet es la red de procesamiento que utilizan las tarjetas Visa. Qué tipo de programación crees que utiliza. Justifica tu respuesta.

Utiliza programación distribuida porque se basa en múltiples servidores interconectados, garantizando velocidad, disponibilidad y resistencia a los fallos.

### Ejercicio 3

Explica los estados y la transición de cada uno de ellos explicando qué sucede cuando se realiza una videollamada. Describe qué ocurre para que el proceso de la videollamada pase de un estado a otro.

- **Nuevo:** El usuario 1 realiza una videollamada al usuario 2 y comienza el proceso.
- **Listo (Ready):** El usuario 2 recibe la videollamada y se espera a que conteste para iniciar la ejecución del proceso.
- **En Ejecución (Running):** Los dos usuarios se encuentran en la videollamada.
- **Bloqueado (Blocked):** Un ejemplo podría ser un fallo en la conexión, depende de factores externos por lo cual quedaría el proceso en pausa.
- **Terminado:** Cuando los dos usuarios cuelgan se finaliza el proceso.



### Ejercicio 4

Observa la siguiente captura de pantalla del comando top y responde a las preguntas:

Fuente: chatgpt - Cuando ejecutas el comando top en linux que significa cada columna que te aparece?

a) ¿Cuál es el PID del proceso spotify?

El PID de spotify es 1630.

b) ¿En qué estado se encuentra el proceso code? ¿Qué significa ese estado?

Se encuentra en estado S (Sleeping), lo cual significa que se encuentra en espera.

c) ¿Qué proceso tiene más prioridad de los tres mostrados del usuario ana? .

Los procesos de ana con mayor prioridad son brave y code ya que tienen la misma PR y NI (20 | 0), a diferencia del proceso de spotify el cual tiene un PR de 25.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1501	ana	20	0	2.5g	250m	100m	R	15.7	5.0	1:15.20	brave
2105	ana	20	0	1.8g	150m	80m	S	0.0	3.0	0:05.10	code
3010	root	-10	-30	500m	50m	25m	S	5.1	1.0	25:40.80	systemd
1630	ana	25	5	1.2g	120m	60m	S	0.0	2.5	0:10.30	spotify

## Apartado 4

Cuando ejecutas una aplicación, el SO crea un Bloque de Control de Procesos (PCB). Describe tres de los elementos de información más importantes que el Kernel guardaría en el PCB de un proceso y explica brevemente su función.

**PID y PPID** → Le asigna un identificador a los procesos.

**Estado del proceso** → Indicar en qué etapa de ejecución está (listo, ejecutando, bloqueado...).

**Registros** → Guarda la información para volver a la ejecución sin perder los datos.

## Apartado 5

Hemos hablado en clase de un "flujo de control secuencial" a "múltiples flujos de control" o hilos. Imagina una aplicación como un procesador de textos (ej. Microsoft Word). Describe al menos dos tareas que podrían ser gestionadas por dos hilos diferentes dentro del mismo proceso de Word.

A partir de tu ejemplo, explica cuál es la relación fundamental entre un proceso y sus hilos.

Dos tareas gestionadas por dos hilos diferentes podrían ser por ejemplo la **interacción del usuario** al escribir en el documento y luego otro hilo que se encargue del **autoguardado** de estos datos a medida que el usuario va escribiendo.

En este caso, Microsoft Word es un único proceso, y dentro de él existen varios hilos que comparten la misma memoria y recursos del proceso, pero cada uno tiene su propio flujo de ejecución. Esto permite que el usuario pueda seguir escribiendo sin interrupciones mientras otro hilo guarda el documento en segundo plano.

