

Nombre y Apellido:..... Curso:

TEORÍA					PRÁCTICA			NOTA
1	2	3	4	5	1	2	3	

TEORÍA: Responda brevemente las siguientes preguntas. Justifique.

- Explique la utilidad de los PCBs en un entorno que soporta KLTs
- Explique cómo funciona la estrategia de **prevención de deadlocks**, basándose en las cuatro condiciones. Indique ventajas y desventajas respecto a la **evasión de deadlocks**.
- V o F Justifique. En las soluciones de hardware para la Exclusión Mutua...
 - Deshabilitar las interrupciones: eficiente en sistemas de multiprocesamiento pero su uso requiere espera activa.
 - Instrucciones especiales: al ser atómicas evitan la espera activa y condiciones de carrera.
- Explique la utilidad del planificador de largo plazo. Indique qué estados del proceso están relacionados con este planificador y qué función cumplen.
- Explique el concepto de inanición (dando ejemplos) aplicado a:
 - Sincronización
 - Planificación
 - Deadlock

PRÁCTICA: Resuelva los siguientes ejercicios justificando las conclusiones obtenidas.

Ejercicio 1

Un sistema que planifica procesos utilizando el algoritmo VRR con Quantum = 3 tiene la siguiente traza de ejecución.

	Llegada	CPU	E/S	CPU	E/S	CPU
P1	1	5	1	1	-	-
P2	2	1	1	1	2	1
P3	0	4	2	5	-	-

- Realice el diagrama de GANTT e indique el orden de los procesos listos en cada instante.
- Indique los cambios de modo que ocurrieron durante la ejecución.

Ejercicio 2

El Manchester City de Guardiola emula su juego vistoso utilizando 9 procesos “jugadores” (3 por cada zona) y un proceso “delantero”. Para ello utilizan una variable compartida PELOTA, que indica de dónde vino la pelota, y es inicializada en “Der” o “Izq” cuando arranca la jugada, Todas las jugadas comienzan con un jugador del centro pasando la pelota hacia un costado, y siguen con los jugadores de los costados pasando la pelota al centro. Después de 30 pases, el delantero busca la pelota sin importar donde esté, y patear al arco. Luego de esto, la jugada se reinicia.

Jugador de la izq (3)	Jugador del centro (3)	Jugador de la der (3)	Delantero (1)
while (1) { pasar(“Centro”); PELOTA = “Izq”; }	while (1) { if(PELOTA == “Der”) { pasar(“Izq”); } else { pasar(“Der”); } }	while (1) { pasar(“Centro”); PELOTA = “Der”; }	while (1) { buscar_pelota(); patear_al_arco(); }

Sincronice los procesos utilizando únicamente semáforos, para que Pep Guardiola pueda aprovechar el entrenamiento, sin caer en inanición, deadlocks o inconsistencias.

Ejercicio 3

Un motor de base de datos que detecta y recupera deadlocks provee la siguiente información:

	R1	R2	R3	R4
P1	1	0	0	0
P2	0	0	0	1
P3	0	1	0	0
P4	0	0	1	0
P5	2	0	0	1

Pedidos

	R1	R2	R3	R4
P1	0	0	0	1
P2	1	0	0	0
P3	0	0	1	0
P4	0	1	0	0
P5	0	0	0	0

Asignaciones

Los recursos disponibles son (0,0,0,0)

- Indique qué procesos se encuentran en deadlock y cómo podría solucionar el problema
- Indique el mínimo de recursos disponibles que debería haber, para que ningún proceso sufra inanición.

Condiciones de aprobación: 3 preguntas correctamente respondidas y 1,5 ejercicios correctamente resueltos.