

# 2021 - Sistemas Operativos General

Área personal / Mis cursos / 2021 - Sistemas Operativos General / Parciales 1C 2021 / 2021 - 1C - 1er Parcial - Práctica - TT

Comenzado el

Saturday, 5 de June de 2021, 15:06

Estado

Finalizado

Finalizado en

Saturday, 5 de June de 2021, 16:16

Tiempo empleado

1 hora 10 minutos

Pregunta 1

Parcialmente correcta

Puntúa como 1,00

⚑ Marcar pregunta

Se dispone de varios pacientes que desean atenderse en un consultorio. Existen dos recepcionistas que atienden en simultáneo (empezando por recepcionista nº1), con la orden de no asignar en total más de tres turnos en simultáneo.

Se desarrolló el siguiente pseudo código:

```
variable_global RESP_ENCUESTA = true;
```

Paciente (N instancias)	Recepcionista 1 (1 instancia)	Recepcionista 2 (1 instancia)
pedir_turno()		
SIGNAL(A)	while(TRUE)	while(TRUE)
WAIT(B)	WAIT(D)	WAIT(E)
atenderse()	WAIT(A)	WAIT(A)
SIGNAL(C)	WAIT(C)	WAIT(C)
SYSCALL1()	asignar_turno()	asignar_turno()
if (RESP_ENCUESTA==true){	SIGNAL(B)	SIGNAL(B)
responder_encuesta();	SIGNAL(E)	SIGNAL(D)
}		
SYSCALL2()		

Se pide responder lo siguiente:

- Los valores iniciales de los semáforos deberían ser A= 0 , B= 0 , C= 3 , D= 1 , E= 0
- Podrían sufrir una situación de starvation las instancias del Proceso: Paciente
- Si las llamadas a WAIT(D) y WAIT(A) en Recepcionista 1 fueran alternadas en su orden (esto es, intercambiadas), esto generaría: Comportamiento similar
- Si las llamadas a SIGNAL(B) y SIGNAL(E) en Recepcionista 1 fueran alternadas en su orden, esto generaría: Posible bloqueo
- Indique que operación debería realizarse en reemplazo de SYSCALL1: WAIT(NUEVO\_SEMAFORO) y SYSCALL2: Ninguno

Navegación por el cuestionario

1

2

3

4

Mostrar una página cada vez

Finalizar revisión

Pregunta 2

Parcialmente correcta

Puntúa como 1,00

⚑ Marcar pregunta

Se desea desarrollar un sistema operativo donde los procesos repartan el tiempo de CPU equitativamente, evitando monopolizar la CPU y evitando que los procesos cpu bound (limitados por cpu) perjudiquen a los i/o bound (limitados por i/o). No debería existir planificación de mediano plazo.

Para garantizar la estabilidad del sistema, en el mismo no debería haber más de 5 procesos en ready, 6 en ejecución y 7 bloqueados. En los estados New/Exit no debería haber más de 7 procesos en cada estado.

El sistema no tendrá KLTs, pero debería permitir el uso de ULTs.

Responda lo siguiente sobre dicho sistema a diseñar:

- Cantidad necesaria de estados, en el diagrama de estados de los procesos: Ready / Running / Blocked / New / Exit
- Algoritmo de planificación adecuado: Multinivel
- Nivel máximo de multiprogramación que tendrá el sistema: 18
- La información sobre planificación para el planificador de corto plazo del sistema operativo (como por ejemplo: prioridad, quantum ó estimación de ráfaga; según se necesite) deberá ser ubicada en el siguiente elemento de la imagen del proceso: Ninguno

Pregunta 3

Correcta

Puntúa como 1,00

⚑ Marcar pregunta

Dado el siguiente diagrama y traza de ejecución de 4 procesos:

Procesos	Arribo	CPU	I/O	CPU
A	0	2	2	5
B	1	5	1	2
C	3	1	1	4
D	0	3	2	3

Como referencia, la línea verde del diagrama se encuentra en el instante 5

Nota: ante simultaneidad en tiempos de llegada, se elige por orden alfabético ascendente.

- a) Determine qué algoritmo de PCP (Planificador de Corto Plazo) está utilizando: VRR
- b) En base al algoritmo seleccionado en el punto 1, determine por qué en el instante 5 selecciona a ese proceso a ejecutar: Ninguna de las opciones es válida
- c) Determine qué proceso continuará ejecutando en el instante 11: A
- d) Determine el orden en que finalizan los procesos: D/A/C/B

Pregunta 4

Incorrecta

Puntúa como 1,00

⚑ Marcar pregunta

Dada la siguiente traza de ejecución, responda las preguntas teniendo en cuenta que el planificador de corto plazo utiliza Round Robin y la biblioteca de hilos de usuario utiliza SJF con desalojo.

Nota: el tamaño del Q se puede deducir

		LL	CPU	IO	CPU
P1	UA	2	1	2	3
	UB	8	1	1	1
	UC	2	4	-	-
P2	KD	0	2	1	1
	KE	1	1	1	4

LL: Tiempo de Llegada

1. ¿En qué instante se produce la última interrupción de clock? 10

2. Si la biblioteca de hilos utilizase jacketing, ¿Cuántas unidades de tiempo consecutivas ejecutaría el siguiente hilo de usuario del proceso 1 a partir del instante 4? 6

3. Si los hilos del proceso 2 fuesen ULT y la biblioteca utilizase jacketing, ¿Cuál sería el siguiente hilo a ejecutar en el sistema a partir del instante 2? ULT A

4. ¿Por qué en el instante 10 ejecuta el ULT B? ULT B realizó una llamada al sistema usando un wrapper de la biblioteca de hilos