



Estudiante:  Valentín

/ UGR / plataforma de apoyo a la docencia

junio

21

22:35



Plataforma

> España

> ugr.es

> ETSIIT

> Db.Gr.Ing.Inf./Matem.

>









AC 

2 notificaciones




Arquitectura de Computadores


AC

- 
Inicio
- 
Asignatura
- 
Evaluación
- 
Archivos
- 
Usuarios
- 
Comunicación
- 
Análisis
- 
Perfil


Frecuentes




Exámenes




Documentos




Calificaciones



Timeline



Test



Compartidos

JUNIO 2021							
L	M	M	J	V	S	D	
31	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30	1	2	3	4	

5 6 7 8 9 10 11

2021-06-15
17:48En
Archivos>Docume
se ha p...

...

Mancia
Anguita López

39 de AC

1 profesor

	Juan Jo... Escoba...	3:18'09"
38 estudiantes		
	Valentín Guerrer...	40"
	Federico Cabrer...	4'07"
	Germá... Padua ...	12'52"
	Alberto Díaz C...	25'48"
	Juan A... Ruiz Ar...	33'05"
	José D... Barran...	34'06"
	Javier Garrue...	38'35"
	Juan Fernán...	55'22"
	Alejandro Cárden...	1:03'47"

...

Sistema Actividades Proyectos Convocatorias Test **Exámenes** Juegos Encuestas

Control Tema 3

Universidad de Granada - Doble Grado en Ingeniería Informática y Matemáticas
Arquitectura de Computadores



Desconocido: 45338112 Guerrero Cano, Valentín



Inicio: 2021-06-02, miércoles, 16:05:09

Final: 2021-06-02, miércoles, 16:31:19

Preguntas: 9

Respuestas
válidas: Puntuación: Nota: 

1 El protocolo MSI utiliza posescritura (write-back).
V/F Usuario Profesores
V **V**

2 El protocolo MESI utiliza escritura con actualización (write-update).
V/F Usuario Profesores
F **F**

3 En un multiprocesador con protocolo MESI, un controlador de caché recibe de un procesador de su nodo una petición de lectura de un bloque que tiene en estado E en su caché (la caché de su nodo), entonces el estado siguiente del bloque en esta caché será de (ponga el nombre completo del estado, no la inicial, no se distingue entre minúsculas y mayúsculas, use "depende" si hay varias posibilidades):
Texto Usuario Profesores
Modificado a) exclusivo

4 En un multiprocesador con protocolo MSI, un controlador de caché recibe de un procesador de su nodo una petición de lectura de un bloque que tiene en estado M en la caché de su nodo (su caché), entonces el estado siguiente del bloque en la caché de su nodo (su caché) será de (ponga el nombre completo del estado, no la inicial, no se distingue entre minúsculas y mayúsculas):
Texto Usuario Profesores
Modificado a) modificado

5 En un placa NUMA se implementa un protocolo MSI de mantenimiento de coherencia basado en directorios sin difusión. Un nodo solicitante envía una petición de acceso a memoria a otro nodo, que es el nodo origen del bloque al que quiere acceder, aunque realmente tiene copia válida del bloque en la caché. ¿Qué paquete de petición ha recibido el nodo origen del solicitante? (use las abreviaturas que hemos usado en clase para los paquetes)
Texto Usuario Profesores
PtLec a) PtEx
b) PtEx(B)
c) PtEx()
d) Petición de acceso exclusivo

6 En un placa NUMA con 4 nodos se implementa un protocolo MSI de mantenimiento de coherencia basado en directorios sin difusión. Cada nodo tiene un trozo de memoria principal de 3 GiB y una caché de 1MiB, los bloques de memoria son de 64 bytes, se usa vector de bits de presencia y 1 bit para codificar el estado de un bloque en la memoria principal ¿Cuántas entradas (filas) tiene el subdirectorio de memoria principal en un nodo? (el directorio está distribuido entre los nodos, directorio en un nodo=subdirectorio)
N.º entero Usuario Profesores

50331648 50331648

7 Si el modelo de consistencia de memoria de un multiprocesador no garantiza el orden W->R significa que una lectura de memoria de un procesador puede adelantar a una escritura en memoria de otro procesador siempre que la lectura y la escritura no accedan a la misma dirección de memoria.
V/F

Usuario Profesores

V

F

8 Para implementar un `lock()` de un cerrojo simple en multiprocesadores que no garantizan el orden W->R se podría usar el siguiente código:
V/F

```
lock(k) {
while (Fetch_and_Or(k,1)==1) {};
}
```

Usuario Profesores

V

V

9 Para implementar un `unlock()` de un cerrojo simple en procesadores con arquitectura ARMv7 basta con usar una instrucción de almacenamiento en memoria.
V/F

Usuario Profesores

V

F

Información DocumentaUGR

CommunitySoftware liAndroid

iOS

¿Qué es SWAD	Manual breve	Condiciones leg	Twitter	Source code	SWADroid	Goog	iSWAD	App St
What is SWAD?	Brief manual	[E	Protección de d	Facebook	Download	SWADroid Blog	iSWAD	Twitter
Publicaciones	Guía usuario	[I	Twitter	SWAD	Wikipedia	Install	SWADroid	Twitter
Funcionalidad	User guide	[E	Estadísticas	Google+	Database	SWADroid	Goog	
Difusión	Presentacion	Póster	YouTube	Translation	SWADroid	GitHub		
Prensa	Videotutoriales	Servidor	alternativeTo	API	SWADroid	Open HUB		
	Logos	Encuentro	startupRANKI	Changelog				
			Capterra	Roadmap				
			SourceForge	Authors				
			GitHub	Implementación				
			Open HUB					



Universidad de Granada

Consultas y problemas: swad@ugr.es

Acerca de SWAD 20.88.2 (2021-06-02) Página generada en 37 ms y enviada en 95 µs