

PREGUNTAS TIPO AC TEMAS 3 Y 4

1. En un NUMA el controlador de memoria se encuentra en un chipset
F
2. En un VLIW las instrucciones que se ejecutan en paralelo se captan juntas de memoria.
V
3. En un VLIW el hardware extrae el paralelismo a nivel de instrucción.
F
4. En un SMT se multiplexa el almacenamiento.
F
5. En un SMT se comparten las unidades funcionales.
V
6. En TMT la conmutación la decide el software.
F
7. Los UMA implementan mantenimiento de coherencia.
F
8. La comunicación de datos entre los procesadores la realiza el sistema de memoria.
V
9. Con datos modificables privados se puede tener incoherencia entre cachés de distinto nivel.
V
10. Se producen menos accesos a caché por medio de la actualización write-through que por write-back.
F
11. Se pueden producir incoherencias con write-through
F
12. Se pueden producir incoherencias con mecanismos de propagación.
V
13. Los protocolos snoopy es mejor hacerlos en sistemas con un número de nodos pequeño.
V
14. Se pueden tener dos nodos con copia válida de un bloque (1.1.0.....) y que dicho bloque esté en estado inválido en memoria.
F
15. En el protocolo MSI basado en directorios con difusión las respuestas se mandan al nodo origen.
V

16. Es posible implementar, con ciertas modificaciones del hardware, la consistencia secuencial.
F
17. Hay algún procesador que permite que otro procesador pueda ver un acceso a memoria antes que todos los demás.
V
18. El buffer de escritura FIFO provoca que las escrituras retarden las lecturas en la ejecución de un programa.
F
19. El código de sincronización entre procesadores asegura que se cumpla el orden de los programas que se ejecutan en cada procesador.
F
20. Si varios procesos acceden al cierre del cerrojo a la vez, más de uno podrá entrar.
F
21. El método de liberación puede resolver la liberación de un thread o de varios.
V
22. La apertura del cerrojo debe ser atómica.
V
23. Las barreras se pueden reutilizar tantas veces como se quiera.
F
24. En OpenMP deberíamos usar atomic en vez de critical.
V
25. El hardware de un procesador superescalar elimina dependencias RAW.
F
26. El hardware de un procesador VLIW elimina dependencias RAW.
F
27. Mediante la predicción adelantada de saltos podemos eliminar dependencias de control.
V
28. En un superescalar podemos evitar dependencias incluyendo instrucciones nop.
F
29. Las instrucciones se captan desde memoria principal
F
30. En la etapa de captación se incluye hardware para predecir saltos.
V
31. La implementación dinámica del renombrado de registros supone añadir circuitería extra.
V
32. El banco de registros de renombrado es compartido por la ventana de instrucciones y por la etapa de captación.
F

33. No se puede paralelizar un programa que no relaje el orden W-R

F

34. La ejecución especulativa no añade ningún tipo de sobrecarga al procesador.

F

35. Una instrucción que se ha ejecutado de manera especulativa y ha resultado ser errónea se puede retirar del ROB.

F

36. En una arquitectura VLIW, el software no puede reducir las dependencias RAW.

F

37. En una arquitectura VLIW se introducen instrucciones nop para evitar dependencias.

V

38. Podemos usar la predicción estática en un procesador con predicción dinámica de saltos.

V

39. Un procesador superescalar depende directamente del compilador.

F