

APELLIDOS: \_\_\_\_\_ NOMBRE: \_\_\_\_\_

Hojas adicionales adjuntas: \_\_\_\_\_

*Normas: Escribe los apellidos y el nombre en cada hoja de examen y en cada hoja adicional que utilices. Es posible escribir la respuesta en la misma hoja de cada ejercicio, pero si utilizas hojas adicionales no mezcles en la misma hoja respuestas de ejercicios diferentes: se entregará cada ejercicio por separado. La respuesta a los ejercicios debe estar escrita con bolígrafo. Está permitido el uso de calculadora. Al terminar, indica en cada hoja de examen las hojas adicionales utilizadas para cada ejercicio.*

Un sistema cuenta con un procesador con una frecuencia de reloj de 4 GHz. El ancho de palabra es de 64 bits. La memoria principal es de 1 GiB ( $2^{30}$  B), divididos en 2 módulos de 512 MiB con entrelazamiento de orden inferior configurados en *Dual Channel*.

Dispone de memoria caché organizada en dos niveles

- L1: 32 kiB ( $2^{15}$  B), líneas de 64 Bytes, asociativa por conjuntos de 8 vías, con algoritmo de reemplazo LRU y políticas de ubicar en escritura (*write-allocate*) y post-escritura (*write-back*)
- L2: 256 kiB ( $2^{18}$  B), líneas de 64 Bytes, asociativa por conjuntos de 4 vías, con algoritmo de reemplazo FIFO y políticas de ubicar en escritura (*write-allocate*) y post-escritura (*write-back*)

El espacio virtual está paginado, con un esquema de traducción en 2 niveles. El tamaño de página es de 4 kiB y cada Tabla de Páginas de primer o segundo nivel ocupa exactamente una página. Cada entrada de una Tabla de Páginas ocupa exactamente 4 Bytes, donde el bit más significativo es el bit de residencia y los bits menos significativos son el número de página física. Los restantes son bits de protección y control. Para acelerar la traducción se utiliza una TLB de 256 entradas.

Como almacenamiento secundario tiene un conjunto de discos SSD de 2 TiB cada uno.

El procesador, la memoria principal y el almacenamiento secundario están conectados a través de un bus multiplexado síncrono de 64 bits a 1 GHz.

Contesta a las siguientes preguntas:

- [2p] 1. ☐ Solicito el examen correspondiente a la parte de procesador. Marcando esta casilla se anulará la calificación obtenida en el primer parcial de ejercicios y se reemplazará por la obtenida en este examen.
- [2p] 2. ☐ Solicito el examen correspondiente a la parte de memoria principal y memoria caché. Marcando esta casilla se anulará la calificación obtenida en el segundo parcial de ejercicios y se reemplazará por la obtenida en este examen.

[0.75p] 3. Un núcleo de un procesador Intel Core i9-12900K trabaja a una frecuencia base de 4.0 GHz, y posee dos unidades de ejecución vectoriales AVX-2 capaces de ejecutar 8 operaciones en punto flotante por ciclo cada una. Contesta a las siguientes preguntas:

- (a) **0.25p** ¿Cuál es el rendimiento máximo teórico en GFLOPS de este núcleo? Sabiendo que el procesador cuenta con ocho núcleos idénticos, ¿cuál será el rendimiento máximo teórico del procesador completo?

$$1 \text{ núcleo} \Rightarrow 4.0 \text{ GHz} \times 2 \text{ VU/núcleo} \times 8 \text{ FLOPS/VU} = 64 \text{ GFLOPS/núcleo}$$

$$1 \text{ procesador} \Rightarrow 64 \text{ GFLOPS/núcleo} \times 8 \text{ núcleos} = 512 \text{ GFLOPS}$$

- (b) **0.25p** Se ejecuta en el procesador el siguiente código:

```
float r, a[N];  
for( int i = 0; i < N; ++i )  
    r = r + a[i];
```

Supongamos que el rendimiento de este código no está limitado por la capacidad de las unidades de ejecución en punto flotante, sino por el sistema de memoria. Si el mismo tiene un ancho de banda de 32 GB/s, ¿cuál será el rendimiento máximo en GFLOPS del procesador al ejecutar el código? Recuerda que cada elemento de tipo `float` ocupa 4 bytes.

Con 32 GB/s de ancho de banda se pueden transferir  $32 \cdot 10^9 / 4 = 8 \text{ Gfloats}$  por segundo. Por cada elemento de tipo `float` transferido podemos ejecutar un único FLOP (la suma). Por tanto, el rendimiento máximo estará limitado a 8 GFLOPS.

- (c) **0.25p** Supongamos ahora que el código ensamblador que ejecuta el lazo del apartado anterior incluye una carga, una suma en punto flotante, una suma de enteros con operando inmediato y un salto condicional. En una ejecución concreta del código se ha medido un rendimiento de 6.4 GFLOPS. ¿Cuál ha sido el rendimiento expresado en MIPS?

Dado que se han obtenido 6.4 GFLOPS y por cada FLOP se ejecutan 4 instrucciones, deben haberse ejecutado  $6.4 \cdot 10^9 \text{ FLOPS} \times 4 \text{ instrucciones/FLOP} = 25.6 \cdot 10^9 \text{ instrucciones/s} = 25600 \text{ MIPS}$ .

Nota: Estas instrucciones se corresponden con un código ensamblador como el siguiente:

```
loop:  
    lwc1 $f0, 0($a0)    # carga  
    add.s $f2, $f2, $f0 # suma en punto flotante  
    addi $a0, $a0, 4    # suma de enteros  
    bne $a0, $a1, loop  # salto condicional
```

- [2p] 4. La CPU solicita los datos correspondientes a la dirección virtual 0x0121 27B4. En la entrada de la tabla de páginas de primer nivel se encuentra el contenido 0xA274 14A0 y en la entrada de la tabla de páginas de segundo nivel 0x8002 4605.

(a) **0.25p** Determina en qué campos se divide la dirección virtual.

El tamaño de página es  $2^{12}$  Bytes, por lo que se necesitan 12 bits para el desplazamiento. Cada TP de primer o segundo nivel tiene  $\frac{2^{12} \text{ Bytes}}{4 \text{ Bytes/entrada}} = 2^{10}$  entradas. Entonces son 10 bits para PV1, 10 bits para PV2 y 12 bits para desplazamiento.

Dirección virtual		
numPV1	numPV2	$\Delta_p$
10 bits	10 bits	12 bits
0x004	0x212	0x7B4

(b) **0.25p** Determina cómo se divide una dirección física para memoria virtual.

El sistema tiene 1 GiB de memoria física ( $2^{30}$  bytes). Como el tamaño de página es de 4 kiB ( $2^{12}$  bytes), hay en total  $2^{30}/2^{12} = 2^{18}$  páginas físicas. La dirección física se divide de la siguiente forma:

Dirección física	
PF	$\Delta_p$
18 bits	12 bits

(c) **0.4p** Calcula la dirección física de las entradas correspondientes de las tablas de páginas de nivel 1 y 2 si la tabla de páginas de primer nivel se encuentra al comienzo de la memoria física.

El registro base de la TP de nivel 1 es 0, como nos indica el enunciado. En esa tabla tenemos que desplazar hasta la entrada 4, correspondiente al número de PV de nivel 1. Cada entrada ocupa 4 Bytes.

- $DirE1 = 0 + 4 \times 4 = 0x10$

Del contenido de esa entrada, que está en el enunciado, obtenemos la página física en la que comienza la tabla de nivel 2 (los 18 bits menos significativos): 0x014A0. El registro base de la tabla es la dirección física en la que comienza esa tabla: 0x014A0000. Partiendo de esa dirección desplazamos 4 Bytes/entrada hasta la entrada 0x212, correspondiente al número de PV de nivel 2.

- $DirE2 = 0x014A0000 + 0x212 \times 4 = 0x014A 0848$

(d) **0.2p** ¿Cuál es la dirección física resultante del proceso de traducción?

La página física son los 18 bits menos significativos de la entrada de la TP2: 0x24605.

A esa página física le concatenamos el desplazamiento 0x7B4.

La dirección física resultante es 0x246057B4.

(e) **0.1p** ¿Qué información se escribirá en la TLB?

Se escribe el par PV-PF de esta traducción realizada: 0x03212 - 0x24605

(f) **0.2p** Determina el tamaño del Espacio Virtual.

Las direcciones virtuales tienen  $10+10+12 = 32 \text{ bits}$ , por lo que el espacio virtual es de  $2^{32} = 4 \text{ GiB}$ .

(g) **0.2p** Si la caché L1 es una caché con índices virtuales y etiquetas físicas (VIPT), calcula en qué conjunto de la caché se encontrará el dato referenciado por la dirección virtual del enunciado.

Las direcciones caché en L1 tienen 6 bits de desplazamiento (el tamaño de línea es de 64 Bytes) y 6 bits de índice (512 líneas en conjuntos de 8 vías  $\rightarrow$  64 conjuntos).

Por tanto los bits de  $\Delta_p$  de la DV,  $0x7B4$  o  $0111\ 1011\ 0100_2$  los dividimos de la siguiente forma:

$\Delta_p$	
Índice	Despl.línea
$011110_2$	$110100_2$

El índice es  $0x1E$  y el desplazamiento  $0x34$ , por lo que el dato se encuentra en el conjunto  $0x1E$ .

(h) **0.2p** ¿Qué ventajas e inconvenientes tiene una caché virtual con respecto a otros tipos de caché?

Las cachés virtuales se indexan por las direcciones virtuales en lugar de físicas, y por tanto permiten acceder a memoria caché directamente sin necesidad de traducir las direcciones de memoria. Solo es necesario realizar la traducción en caso de fallo caché. Sin embargo, necesitan información adicional de cada proceso en el directorio caché o bien purgar la caché completa en cada cambio de contexto para evitar problemas graves de seguridad.

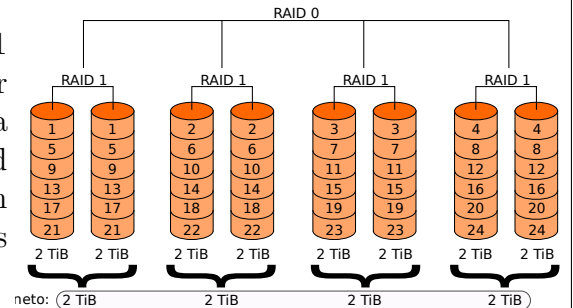
(i) **0.2p** ¿Qué ventaja aporta realizar la traducción en varios niveles con respecto a una traducción directa en un único nivel?

Se reduce considerablemente el espacio necesario para almacenar las tablas de páginas en memoria principal, porque se mantienen en memoria física solo aquellas secciones de la tabla de páginas que corresponden al espacio virtual utilizado.

- [0.5p] 5. El almacenamiento secundario de este sistema está configurado en RAID 1+0 para incrementar la seguridad de la información.

(a) **0.15p** ¿Cuántos discos reales tiene el sistema para la capacidad neta de 8 TiB?

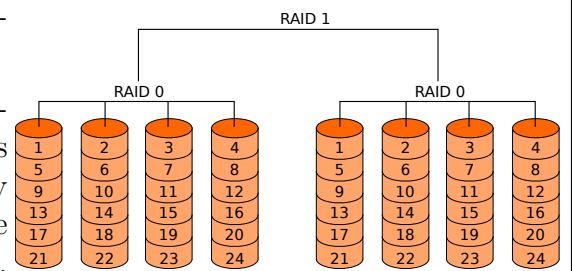
La redundancia en RAID 1+0 la proporciona el RAID 1 (mirroring) sobre el que aplicamos el RAID 0 (striping). Por cada disco de información neta existe un disco espejo con la misma información, por lo que para conseguir la capacidad neta de 8 TiB, que serían 4 discos de 2 TiB, se requieren 8 TiB adicionales con información redundante. Necesitamos en total **8 discos** de 2 TiB.



(b) **0.20p** ¿Qué ventajas o inconvenientes tendría sobre el método actual utilizar RAID 0+1?

RAID 0+1 es un RAID 1 montado sobre un RAID 0. Mantiene la misma cantidad de información redundante que RAID 1+0. Sin embargo, según el controlador RAID que se utilice pueden variar las condiciones que permiten reconstruir el RAID en caso de fallo simultáneo de discos.

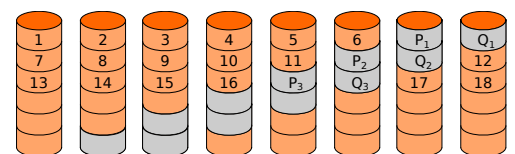
Estos sistemas son capaces de recuperarse ante fallos de discos siempre y cuando no fallen simultáneamente dos discos del mismo RAID 1. En el caso de RAID 1+0, cuando hay un fallo de disco, solo el fallo simultáneo en uno de los siete restantes (en su espejo) produciría un error irreparable. En cambio, en la misma situación en un RAID 0+1, si el fallo simultáneo ocurre en alguno de los cuatro discos que conforman el espejo el sistema fallaría.



(c) **0.15p** Para incrementar la cantidad de información neta en el sistema sin invertir en nuevo hardware, garantizando a la vez que exista cierta redundancia, se implementa un RAID 6. ¿Cuál sería ahora la cantidad de información neta que tendríamos?

Un RAID 6 incorpora doble paridad distribuida, por lo que la información redundante ocupa un tamaño equivalente al de dos unidades de disco.

En este caso, como disponemos de 8 discos de 2 TiB, la **capacidad neta es de 12 TiB** (6 discos) y la información redundante es 4 TiB.



[0.75p] 6. A través del bus indicado, la CPU solicita la transferencia de una página (4 kiB) a memoria física. En cada ciclo de reloj del bus (frecuencia 1 GHz) se pueden transmitir 64 bits de información o una dirección de memoria. El bus permite transacciones de 32 palabras, necesitando para el acceso al primer bloque de 8 palabras en el sistema de almacenamiento 100 ns, y para cada grupo de 8 palabras adicional 30 ns. La transferencia de un bloque a través del bus se puede solapar con el acceso a los siguientes datos. Entre transacciones de bus es necesario esperar 1 ciclo de reloj de bus.

(a) **0.25p** ¿Cuántas transacciones de bus se realizarán para transferir una página completa?

$$1 \text{ transacción} = 32 \text{ palabras} \times 8 \text{ bytes/palabra} = 256 \text{ bytes} = 2^8 \text{ bytes}$$

$$\frac{2^{12} \text{ bytes/página}}{2^8 \text{ bytes/transacción}} = 16 \text{ transacciones/página}$$

(b) **0.50p** Calcula la latencia de la transferencia completa.

El acceso a datos se realiza en bloques de 8 palabras de 64 bits cada una. Como la transacción comprende 32 palabras, es necesario transferir 4 bloques.

Como la frecuencia del bus es 1 GHz, el tiempo de ciclo es  $1/10^9 = 1 \text{ ns}$ .

Una transacción tiene los siguientes pasos:

- Comienza con el envío de la dirección de memoria (1 ciclo).
- El acceso al primer bloque tarda 100 ns, o 100 ciclos.
- A continuación se transfieren las 8 palabras a razón de 1 palabra/ciclo: 8 ciclos en total.
  - A la vez que se transfieren estas palabras, se accede al siguiente bloque en 30 ns o 30 ciclos.
  - Como el acceso al segundo bloque tarda más que la transferencia del primero, éste será el tiempo que debemos esperar antes de poder transferir el siguiente bloque.
- Esta operación solapada se repite **3 veces**, hasta que se accede al cuarto y último bloque.
- Finalmente se transfiere este último bloque (8 ciclos) y se espera 1 ciclo antes de poder continuar con la siguiente transacción.

$$t_{\text{transacción}} = 1 + 100 + 3 \times \max\{8, 30\} + 8 + 1 = 200 \text{ ns}$$

La latencia de la transferencia de una página es, por tanto:

$$t_{\text{página}} = 16 \text{ transacciones} \times 200 \text{ ns/transacción} = 3200 \text{ ns} = 3.2 \mu\text{s}$$