



به موارد زیر توجه کنید:

- ۱- حتما نام و شماره دانشجویی خود را روی پاسخ نامه بنویسید.
- ۲- کل پاسخ تمرینات را در قالب یک فایل pdf با شماره دانشجویی خود نام گذاری کرده در سامانه CW بارگذاری کنید.
- ۳- این تمرین ۶۰ نمره دارد که معادل ۰,۶ نمره از نمره کلی درس است.
- ۴- در صورت مشاهده هر گونه مشابهت نامتعارف هر دو (یا چند) نفر کل نمره این تمرین را از دست خواهند داد.

۱- (۱۰ نمره) در یک پردازنده با یک حافظه نهان ایده آل ( $miss\ rate=0$ ) اجرای هر دستور یک چرخه ساعت طول می کشد. ( $CPI=1$ ) اما در عمل نرخ فقدان ( $miss\ ratio$ ) حافظه های نهانی که برای داده و دستور به کار می روند ۰,۲ است.

الف- اگر بدانیم دسترسی به حافظه اصلی ۱۰۰ چرخه طول می کشد و ۳۰٪ دستوراتی که روی این پردازنده اجرا می شوند، دستورات load و store هستند، CPI متوسط چقدر است؟

ب- برای افزایش کارایی، از یک لایه دیگر حافظه نهان با زمان دسترسی ۲ چرخه ساعت و نرخ برخورد (hit ratio) ۰,۷۲ برای دسترسی به دستورات استفاده می کنیم. در این صورت CPI متوسط چقدر می شود؟ دقت کنید، در این پردازنده حافظه های نهان داده و دستور کاملاً از هم مجزا هستند.

پاسخ:

الف- به ازای هر دستور یک دسترسی به حافظه دستور داریم. علاوه بر این، در ۳۰٪ دستورات یک دسترسی به حافظه داده داریم.

$$CPI_{avg} = 1 + 0.2 \times 100 + 0.3 \times 0.2 \times 100 = 1 + 20 + 6 = 27\ cycle$$

ب-

$$CPI_{avg} = 1 + 0.2 \times (2 + 0.28 \times 100) + 6 = 1 + 6 + 6 = 13\ cycle$$

۲- (۳۰ نمره) امیرحسین به تازگی علاقه مند شده تاثیر اشتراک پذیری (associativity) را بر عملکرد حافظه نهان به طور کامل درک کند. به این منظور، امیرحسین دو حافظه نهان کوچک را در نظر می گیرد: یک حافظه نهان با نگاشت مستقیم (direct mapped) با ۸ خط و هر خط ۳۲ بیت، و یک حافظه نهان مجموعه-مشارکتی ۴ راهه (4-way associative) با همان اندازه و اندازه خط. برای حافظه مجموعه-مشارکتی، امیرحسین دو سیاست جایگزینی را امتحان می کند، LRU و FIFO.

الف- امیرحسین با دسترسی به دنبال دنباله ای از آدرس های بایت هگزادسیمال زیر، با شروع از حافظه های خالی، حافظه را تست می کند. فرض کنید که آدرس ها تنها ۱۲ بیتی هستند. جداول زیر را با پر کردن مقادیر برجسب (tag) هگزادسیمال برای حافظه و نمایش پیشرفت محتوای حافظه نهان هنگام دسترسی تکمیل کنید. در این جداول 'inv' به معنای نامعتبر بودن محتوای ورودی است. برای سادگی، فقط وقتی عناصری را در جدول پر کنید که مقداری تغییر کرده است.

ب- آیا کمتر بودن اعداد به دست آمده در یکی از جداول نشان از بهتر بودن بی‌چون و چرای آن روش دارد؟ اگر بله، استدلال کنید و اگر خیر، یک سلسله دسترسی بسیار کوتاه به عنوان مثال نقض بیاورید.

ج- تعداد بیت هر کدام از دو نوع حافظه نهان با نگاشت مستقیم و مجموعه مشارکتی ۴ راهه را به دست آورید.

پاسخ:

الف-

Address: 12 bits

Tag: 4 bits [11:8]

Index: 3 bits [7:5]

Offset: 5 bits [4:0]

<u>D-map</u>	Addresses and tags are in HEX								
	line in cache (tag)								hit?
Address	L0	L1	L2	L3	L4	L5	L6	L7	
11B	1	inv	inv	inv	inv	inv	inv	inv	no
134		1							no
20D	2								no
1A2						1			no
105	1								no
360				3					no
27D				2					no
121		1							yes
1A3						1			yes
17A				1					no
307	3								no
273				2					no
131		1							yes

Address: 12 bits

Tag: 6 bits [11:6]

Index: 1 bits [5:5]

Offset: 5 bits [4:0]

<u>4-way</u>	LRU – addresses and tags are in HEX								
Address	line in cache								hit?
	Set 0				Set 1				
	way0	way1	Way2	way3	way0	way1	way2	way3	
11B	4	inv	inv	inv	inv	inv	inv	inv	no
134					4				no
20D		8							no
1A2						6			no
105	-								yes
360							D		no
27D								9	no
121					-				yes
1A3						-			yes
17A							5		no
307			C						no
273								-	yes
131					-				yes

4-way Address	FIFO -- addresses and tags are in HEX								hit?
	line in cache (tag)								
	Set 0				Set 1				
	way0	way1	way2	way3	way0	way1	way2	way3	
11B	4	inv	inv	inv	inv	inv	inv	inv	no
134					4				no
20D		8							no
1A2						6			no
105	-								yes
360							D		no
27D								9	no
121					-				yes
1A3						-			yes
17A					5				no
307			C						no
273								-	yes
131						4			no

	Direct-Mapped	4-way LRU	4-way FIFO
Total Misses	10	8	9
Total Accesses	13	13	13

ب-

حافظه نهان set-associative با جایگزینی LRU بهتر از حافظه نهان مستقیم (direct-mapped) از نظر متوسط تاخیر دسترسی به حافظه است. برای مثال فوق، نرخ فقدان LRU کمی کمتر از نرخ فقدان FIFO است. این به دلیل این است که در دسترسی دهم، سیاست FIFO بلوک tag {4} را به جای tag {D} جایگزین کرد؛ زیرا بلوک {4} بیشتر در حافظه نهان بوده است، حتی اگر بلوک {D} کمترین بار از آن استفاده شده باشد. در این حالت، سیاست LRU از temporal locality بهتری بهره برد.

LRU همیشه از FIFO بهتر نیست. فرض کنید یک حافظه نهان set-associative با پارامترهای مشابه داشته باشیم و یک دنباله دسترسی نشان داده شده زیر را داشته باشیم. برای دسترسی آخر LRU یک miss دارد در حالی که FIFO یک hit دارد.

0x11B, 0x134, 0x20D, 0x1A2, 0x105, 0x360, 0x27D  
0x121, 0x1A3, 0x17A, 0x307, 0x273, 0x361

ج-

در حافظه با دسترسی مستقیم، ۸ خط داریم، بنابراین سه بیت برای اندیس صرف می‌شود. چون هر خط ۳۲ بایت دارد، پنج بیت برای byte offset استفاده می‌شود. پس کلاً ۴ بیت برای برچسب باقی می‌ماند.

$$\text{bitsPerLine} = 1 (\text{valid}) + 4 (\text{tag}) + 32 \times 8 (\text{data}) = 261 \text{ bit}$$

$$\text{TotalBits} = 281 \times 8 = 2088 \text{ bits}$$

در حافظه مجموعه-مشارکتی ۴ راهه، چون اندازه حافظه تغییر نکرده، دو مجموعه خواهیم داشت، بنابراین یک بیت برای اندیس صرف می‌شود و همان پنج بیت برای byte offset و ۲۶ بیت برای برچسب باقی می‌ماند.

$$\text{bitsPerLine} = 4 \times [1 (\text{valid}) + 6 (\text{tag}) + 32 \times 8 (\text{data})] = 1052 \text{ bit}$$

$$\text{TotalBits} = 1052 \times 2 = 2104 \text{ bits}$$

۳- (۲۰ نمره) تعیین کنید هر کدام از تغییرات زیر در حافظه نهان چه تاثیری در هر کدام از ستونها خواهد داشت. در برخی موارد نوع چینش حافظه نهان (مثلا این که نگاشت مستقیم یا کاملا اشتراک پذیر (associative) باشد) روی کم و زیاد شدن برخی پارامترها تاثیر متفاوتی دارد، این موارد را هم مشخص کنید.

Miss penalty	Miss rate	Hit time	Capacity Misses	Conflict Misses	Compulsory Misses
کاهش خطوط کوچک تر می توانند سریع تر وارد شوند. یا تأثیری ندارد چون پهنای باند حافظه از طول اولیه خطوط بیشتر بوده	احتمالا کم می شود چون تاثیر کاهش فقدان های ناشی از تداخل بیشتر است	ممکن است زیاد شود، چون تعداد انتخاب ها بیشتر شده	تغییر نمی کند	کاهش تعداد انتخاب ها بیشتر است، پس امکان تداخل کم می شود	افزایش در هر انتقال بلوک به حافظه نهان، تعداد بایت کمتری وارد می شود
کاهش (به همان دلیل بالا)	احتمالا کم می شود چون تاثیر کاهش فقدان های ناشی از کمبود ظرفیت بیشتر است.	کاهش چون ظرفیت حافظه نهان کم شده	کاهش	تغییر نمی کند	افزایش (به همان دلیل بالا)
تغییر نمی کند	با توجه به توضیحات مربوط به انواع فقدان ها خیلی معلوم نیست	ممکن است زیاد شود، چون تعداد انتخاب ها بیشتر شده	از یک طرف تعداد انتخاب ها بیشتر شده، پس فقدان ناشی از تداخل کم می شود اما از طرفی تعداد مجموعه ها نصف شده پس امکان تداخل بیشتر می شود و می توان گفت یا فقدان ناشی از تداخل یا فقدان ناشی از ظرفیت بیشتر می شود.	تغییر نمی کند	دو برابر کردن اندازه مجموعه ها (ظرفیت و طول خطوط ثابت) چون ظرفیت و طول خطوط ثابت است، تعداد مجموعه ها نصف می شود. از طرفی تعداد انتخاب ها دو برابر می شود
تغییر نمی کند	با توجه به توضیحات مربوط به انواع فقدان ها خیلی معلوم نیست	ممکن است کم شود، چون تعداد انتخاب ها کم شده	دقیقا برعکس بالا از طرفی فقدان ناشی از تداخل بیشتر و از طرفی فقدان ناشی از ظرفیت کمتر می شود	تغییر نمی کند	اگر تعبیر این باشد که تعداد مجموعه ها دو برابر می شود، پس تعداد انتخاب ها نصف خواهد شد
					دو لایه حافظه نهان را باهم ترکیب کنیم و ظرفیت هر دو را برابر همان L1 قرار دهیم. (اندازه خطوط و تعداد مجموعه ها ثابت است)

کاهش	تغییر نمی کند	تغییر نمی کند	تغییر نمی کند	تغییر نمی کند	تغییر نمی کند	تغییر نمی کند
تغییر نمی کند	کاهش	بیشتر می شود چون حافظه نهان بزرگتر شده	در مجموع کم می شود، چون هم تعداد انتخاب ها دو برابر شده است و هم ظرفیت	تغییر نمی کند	تغییر نمی کند	(به معنای اضافه کردن یک لایه حافظه نهان) به معنای دو برابر کردن ظرفیت که اگر تعداد مجموعه ها و اندازه خطوط ثابت باشد، یعنی تعداد انتخاب ها دو برابر می شود