



## تمرین سری هشتم

- پرسش‌های خود را در سامانه CW و تالار مربوط به تمرین مطرح نمایید.
- پاسخ سوالات را تایپ نمایید.
- پاسخ تمرین را به صورت یک فایل زیپ با فرمت HW8\_401234567.zip آپلود کنید. فایل زیپ باید به صورتی باشد که پس از باز کردن آن بدون هیچ پوشه‌ای فایل‌های زیر با ساختار زیر قرار گرفته باشند:

```
1 .  
2 |-- HW8T_401234567.pdf  
3 |-- practical  
4   |-- HW8P_401234567_401234568.pdf  
5   |-- code  
6   |-- ...
```

- در صورت عدم تطابق فایل آپلود شده با فرمت بالا، تمرین شما تصحیح نخواهد شد.
- پاسخ سوالات تئوری و گزارش تمرین‌های عملی باید به فرمت pdf باشد.
- هر دانشجو می‌تواند حداکثر سه تمرین را با دو روز تأخیر بدون کاهش نمره ارسال نماید.
- تمرینات عملی به صورت گروه‌های دو نفر تحویل داده شود.
- هر دو عضو گروه موظف هستند تمرینات خود را بارگذاری کنند.
- عواقب عدم تطابق بین پاسخ دو عضو گروه برعهده خودشان است.
- تحویل تمرین به صورت انگلیسی مجاز نیست. در صورت تحویل تمرین به صورت انگلیسی (حتی بخشی از تمرین) نمره تمرین موردنظر صفر در نظر گرفته می‌شود.
- در صورت مشاهده تقلب برای بار اول نمره هر دو طرف صفر می‌شود. در صورت تکرار نمره کل تمرینات صفر خواهد شد.
- استفاده از ابزارهایی مانند ChatGPT به منظور ابزار کمک آموزشی مجاز است به شرط آن که به خروجی آن اکتفا نشود.
- توجه شود که پروژه نهایی درس در گروه‌های چهار نفر تحویل گرفته می‌شود.
- سوالات با عنوان اختیاری نمره‌ای ندارند اما جواب دادن به آن‌ها کمک به‌سزایی در یادگیری درس می‌کند.

## تمارین تئوری

۱. (آ) یک پردازنده دارای حافظه نهان ۶۴ کیلوبایتی با اندازه بلوک ۶۴ بایتی است. فرض کنید که اندازه حافظه اصلی ۱۶ مگابایت است. قالب آدرس دهی برای دسترسی به حافظه نهان را در دو حالت زیر مشخص کنید (پردازنده byte-addressable است):

- direct mapped
- 2-way associative

(ب) فرض کنید یک پردازنده دارای حافظه نهان سلسله‌مراتبی در دو سطح است. CPI پایه‌ای این پردازنده برابر با ۱.۵ است. نرخ از دست دادن<sup>۱</sup> در حافظه نهان سطح اول برابر با ۸ درصد و جریمه از دست دادن برابر با ۲۰ چرخه ساعت است. نرخ از دست دادن در حافظه نهان سطح دوم برابر با ۲ درصد و جریمه از دست دادن برابر با ۱۰۰ چرخه ساعت است. حال فرض کنید برنامه‌ای روی این پردازنده اجرا می‌شود که ۴۰ درصد دستورات این برنامه، دسترسی به حافظه هستند. CPI پردازنده را هنگام اجرای این برنامه محاسبه کنید.

۲. در طراحی یک حافظه نهان با نگاشت مستقیم<sup>۲</sup> و آدرس‌های ۳۲ بیتی، بیت‌های ۳۱ تا ۱۰ برای برچسب<sup>۳</sup> ۳، ۹ تا ۵ برای اندیس، و ۴ تا ۰ برای آفست استفاده شده‌اند. با توجه به این ساختار به سوالات زیر پاسخ دهید:

(آ) اندازه هر بلوک (خط) چقدر است؟ برحسب کلمات ۳۲ بیتی بگویید.

(ب) تعداد درایه‌های<sup>۴</sup> حافظه نهان چند است؟

(ج) چه درصدی از کل بیت‌های حافظه نهان برای ذخیره داده استفاده شده است؟ (مطابق اسلاید نهم درس)

فرض کنید از زمانی که حافظه نهان کاملاً خالی است آدرس‌های زیر به ترتیب از چپ به راست، درخواست شده‌اند. هر عدد آدرس بایت مورد درخواست است.

0, 4, 16, 132, 232, 160, 1024, 30, 140, 3100, 180, 2180

به سوالات زیر پاسخ دهید:

(د) در هر مورد برچسب، اندیس، آفست، و وقوع برخورد یا فقدان را مشخص کنید. از یک ساختار جدول استفاده کنید.

(ه) نرخ برخورد چقدر است؟

۳. فرض کنید حافظه‌ای ۶۴ کیلوبایتی با بلوک‌های ۱۶ کلمه‌ای داریم که هر کدام از کلمات ۲ بایت هستند. (حافظه مذکور به صورت بایتی آدرس‌پذیر و دارای ساختار little-endian است.)

فرض کنید قطعه کد زیر زده شده است که به صورت کلمه به کلمه بر روی بازه‌ای از حافظه حرکت کرده و در صورتی که مقدار a در کلمه فعلی باشد، مقدار موجود در کلمه‌ای از حافظه که از خانه a در حافظه شروع می‌شود را با حاصل فعلی جمع می‌زند و در نهایت این مقدار را خروجی می‌دهد.

```
sum = 0
for address in range(start, end, 2):
    sum += read_value_from_memory(read_value_from_memory(address))
```

فرض کنید یک حافظه نهان با ۸ بلوک که به صورت direct-mapped است داریم.

فرض کنید مقادیر موجود در حافظه با شروع از بایت ۰ ام و به صورت بایت به بایت از چپ به راست به صورت زیر باشند: (مقادیر موجود دارای مبنای ۱۶ هستند!)

$32_h - 4A_h - 30_h - B8_h - 32_h - B8_h - 34_h - B8_h - 22_h - B8_h - 32_h - 4A_h - 1A_h - 31_h - 42_h - 31_h$

miss rate<sup>۱</sup>  
direct-mapped<sup>۲</sup>  
tag<sup>۳</sup>  
cache entries<sup>۴</sup>

با فرض اینکه کد مذکور با مقادیر  $start = 0$  و  $end = 16$  شروع به اجرا کرده باشد، نرخ برخورد را تا زمان محاسبه خروجی بدست آورید.

۴. کد اسمبلی MIPS زیر یک رشته‌ی کاراکتری که با null خاتمه یافته است را از یک محل به محل دیگر کپی می‌کند.

```
# $t0 has the address of the string source
# $t1 has the address of the string destination
StrCpy:
    lb $t2, 0($t0)    # $t2 gets src char
    sb $t2, 0($t1)    # store into dest char
    beq $t2, $zero, Exit # done when last char is null
    addiu $t0, $t0, 1 # point to next char of src
    addiu $t1, $t1, 1 # point to next char of dest
    j StrCpy
Exit:
```

فرض کنید پردازنده بدون پایپ‌لاین باشد و دارای یک حافظه نهان یکپارچه (داده و دستورالعمل) باشد که به صورت نگاشت مستقیم<sup>۵</sup> با ۱۲۸ بایت ظرفیت و بلوک‌هایی به اندازه‌ی یک کلمه است. فرض کنید حافظه نهان write-through و no-write-allocate است. کد StrCpy در آدرس حافظه 0 قرار دارد و رشته‌های مبدا و مقصد به ترتیب در آدرس‌های  $128_{10}$  و  $256_{10}$  قرار دارند.

- (آ) برای یک بار اجرای حلقه که یک کاراکتر غیر null را کپی می‌کند، چند دسترسی به حافظه انجام می‌شود؟
- (ب) برای یک بار اجرای حلقه که کاراکتر null را کپی کرده و از حلقه خارج می‌شود، چند دسترسی به حافظه انجام می‌شود؟
- اگر فرض کنیم طول رشته‌ی مبدا ۴ بایت (۳ کاراکتر و یک کاراکتر null) باشد، به قسمت‌های بعدی پاسخ دهید.
- (ج) کل تعداد دسترسی‌های حافظه برای اجرای این کد چقدر است؟
- (د) نرخ خطای حافظه نهان را محاسبه کنید.
- (ه) اگر حافظه نهان را به یک حافظه‌ی 2-way set associative با همان ظرفیت و اندازه بلوک تغییر دهیم، نرخ خطای جدید چقدر خواهد بود؟ (فرض کنید جایگزینی به صورت تصادفی صورت می‌گیرد.)
- (و) اگر حافظه‌ی نهان را به یک حافظه‌ی 2-way set associative با write-back و write-allocate تغییر دهیم، نرخ خطا چقدر خواهد بود؟

با توجه به مفاهیم حافظه نهان و فرمت آن که به صورت زیر است، به سوالات زیر پاسخ کامل دهید.

Tag	Index	Block Offset
-----	-------	--------------

۵. (آ) یک حافظه نهان 2-way set-associative در نظر بگیرید که از ۱۰ بیت برای index استفاده می‌کند و بلاک‌های کش آن ۳۲ بایت هستند. اگر ماشین از آدرس‌های فیزیکی ۳۲ بیتی استفاده کند، موارد زیر را حساب کنید:

- تعداد بلاک‌ها در حافظه
- اندازه Block offset
- اندازه Tag

(ب) با توجه به اینکه حافظه از الگوریتم LRU استفاده می‌کند و مجموعاً شامل ۴ بلاک است که هر بلاک ۴ بایت دارد، کدام یک از دسترسی‌های زیر به بایت‌ها برخورد دارند؟

<sup>۵</sup>direct-mapped

Address	Hit/Miss
110001	
100111	
001111	
001100	
010001	
110010	
100101	
001110	
100001	
110001	

(ج) یک برنامه را در نظر بگیرید که ترکیب دستورات آن در جدول پایین نشان داده شده است. اگر این برنامه روی یک ماشین با خط لوله بدون خط اجرا شود، بدون خطاهای حافظه نهان دستور، نرخ برخورد داده ۹۸٪ می باشد و جریمه ی خطای حافظه نهان برابر با ۱۰۰ چرخه باشد، مقدار CPI این برنامه چقدر خواهد بود؟ توجه کنید که همه ی خطاهای حافظه نهان باعث توقف می شوند.

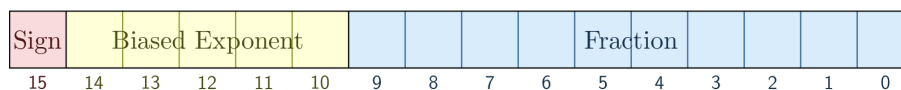
Instruction Type	Frequency
ALU	40%
Loads	25%
Branches	20%
Stores	15%

۶. (آ) **اختیاری** - ساختار هرکدام از حافظه های زیر را به اختصار توضیح دهید. (در توضیحات خود نیازی نیست اجزا را ریز به ریز توضیح دهید. برای مثال، نیازی به توضیح ساختار ترانزیستورهای دروازه شناور نیست.)

- NAND Flash
- NOR Flash
- DRAM

(ب) **اختیاری** - در میان این سه نوع حافظه، کدام یک پرتراکم ترین و کدام یک سریع ترین است؟ دلیل این موضوع را در حد چند کلمه توضیح دهید.

۷. **اختیاری** - استاندارد IEEE 754 در سال ۲۰۰۸ به روز شد تا در کنار فرمت single-width، double-width و سایر فرمت ها، یک فرمت ممیز شناور نیم بندی (half-width) را نیز شامل شود. این فرمت به صورت زیر تعریف می شود:



- طول کل عدد فقط ۱۶ بیت است.
- نما شامل ۵ بیت است که نتیجه می شود bias برابر با ۱۵-.
- بخش کسری شامل ۱۰ بیت است.

واضح است که این half-width نسبت به سایر استانداردها کاراتر بوده و حافظه کمتری مصرف می‌کند، اما دقت اعداد در آن کمتر است. این فرمت عمدتاً در سخت‌افزارهای گرافیکی (مانند کارت‌های تولیدی Nvidia) به کار می‌رود. سؤال‌های زیر (۱ تا ۳) از این نوع داده استفاده می‌کنند.

نکته: اگر ماشین حساب یا تلفن شما توانایی انجام حساب دودویی اعشاری را ندارد، لطفاً از یک ماشین حساب آنلاین استفاده کنید تا حل این تمرین ساده‌تر شود.

(آ) محاسبه  $1 + (-16,360 + 16,360)$  و همچنین  $(16,360 + 1) - 16,360$  را با استفاده از half floats انجام دهید. آیا خاصیت شرکت‌پذیری<sup>۷</sup> در اینجا برقرار است؟

پاسخ مثال:

$$\begin{aligned} (-16,360 + 16,360) + 1 &= 1.0 \quad \underbrace{(0 \ 01111 \ 0000000000)}_{1.0} \bullet \\ -16,360 + (16,360 + 1) &= 0.0 \quad \underbrace{(0 \ 00000 \ 0000000000)}_{0.0} \bullet \end{aligned}$$

بنابراین، برقرار نیست.

(ب) محاسبه  $(0.00048828125 \times 1768) \times 250.125$  و همچنین  $0.00048828125 \times (1768 \times 250.125)$  را با استفاده از half floats انجام دهید. آیا خاصیت شرکت‌پذیری (در اینجا برای ضرب) برقرار است؟

پاسخ مثال:

$$\begin{aligned} (0.00048828125 \times 1768) \times 250.125 &= 215.875 \quad \underbrace{(0 \ 10110 \ 1010111111)}_{215.875} \bullet \\ 0.00048828125 \times (1768 \times 250.125) &= \infty \quad \underbrace{(0 \ 11111 \ 0000000000)}_{\infty} \bullet \end{aligned}$$

نتیجه: برقرار نیست.

(ج) محاسبه  $0.15234375 \times (0.20703125 + 99.6875)$  و همچنین  $(0.15234375 \times 0.20703125) + (0.15234375 \times 99.6875)$  را با استفاده از half floats انجام دهید. آیا خاصیت «توزیع‌پذیری»<sup>۸</sup> برقرار است؟

پاسخ مثال: هر دو عبارت برابر با 15.21875 خواهند بود:

$$0.15234375 \times (0.20703125 + 99.6875) = 15.21875 \quad \underbrace{(0 \ 10010 \ 1110011100)}_{15.21875},$$

$$(0.15234375 \times 0.20703125) + (0.15234375 \times 99.6875) = 15.21875 \quad \underbrace{(0 \ 10010 \ 1110011100)}_{15.21875}.$$

بنابراین، توزیع‌پذیری برقرار است.

۸. اختیاری - به سوالات زیر پاسخ دهید:

(آ) عملکرد کلی DMA را توضیح دهید و نقش آن در کاهش بار پردازنده را شرح دهید.

(ب) در مورد انواع حالت‌های انتقال در DMA تحقیق کنید و تفاوت‌های آن‌ها را شرح دهید:

- Burst Mode
- Cycle Stealing Mode
- Transparent Mode

- (ج) تفاوت بین Memory-Mapped I/O و DMA را از نظر عملکرد سیستم و دخالت پردازنده بررسی کنید.
- (د) تاثیر استفاده از DMA را بر عملکرد کلی سیستم در برنامه‌هایی با نرخ بالای تبادل داده (مانند پردازش ویدیو یا صدا) تحلیل کنید. در پاسخ خود به مواردی مانند پهنای باند، latency و مصرف منابع پردازنده توجه کنید.
- (ه) تحقیق کنید که در سیستم‌هایی با چند وسیله جانبی (I/O devices) چگونه کنترل دسترسی DMA به حافظه مدیریت می‌شود. نقش DMA Controller و اولویت‌بندی در این فرآیند چیست؟
- (و) عملکرد Bus Arbitrator چیست و چه زمانی وارد عمل می‌شود؟
- (ز) راجع به روش‌های arbitration زیر تحقیق کنید.
- Fixed Priority
  - Round Robin
  - Daisy Chaining
- (ح) تاثیر سیاست‌های arbitration بر عملکرد سیستم چیست؟

## تمارین عملی

۱. در این تمرین قصد داریم با استفاده از ابزار ChampSim تنظیمات بهینه برای یک trace را بدست آوریم. در این تمرین از trace موجود در این لینک ، با نام `cassandra_phase0_core0.trace.xz` استفاده کنید. گام‌های زیر برای انجام تمرین لازم است انجام شود:

۱. در ابتدا لازم است که سیاست‌های جایگزینی حافظه نهان زیر را به این ابزار اضافه کنید (ممکن است بعضی سیاست‌ها از قبل موجود باشند):

- Most Recently Used (MRU)
- Least Recently Used (LRU)
- Random

۲. حال مقداری را برای آرگومان‌های `--warmup_instructions` و `--simulation_instructions` به گونه‌ای انتخاب کنید که کل زمان اجرای هر بار شبیه‌سازی حدود ۴ دقیقه باشد و نسبت تعداد دستورات بازه ابتدایی به کل دستورات ۱ به ۴ باشد.

۳. به مقادیر موجود در بخش LLC در فایل `champsim_config.json` توجه کنید. در این بخش مقادیر `way` و `set` نحوه پیاده‌سازی حافظه نهان را نشان می‌دهند. شما باید با تغییر این مقادیر، بهترین ترکیب سیاست جایگزینی با مقادیر `set` و `way` را به جهت بیشینه کردن نرخ برخورد پیدا کنید. به این منظور به نکات زیر توجه کنید:

- در هر مرحله آزمایش اندازه بلوک‌های را ثابت در نظر بگیرید. همچنین حاصل ضرب `set` و `way` باید عدد ثابتی باشد.
- لازم است آزمایش را برای مقادیر ۳۲، ۱۶، ۸، ۴، ۲ و ۱ برای متغیر `set` انجام دهید. (در مجموع ۳۰ مرتبه شبیه‌سازی)
- مقادیر گزارش شده در پایان شبیه‌سازی را برای LLC بررسی کنید.

۴. در نهایت نتایج را به صورت یک نمودار از نرخ برخورد برای هر سیاست با مقادیر مختلف `set` نشان دهید. در حد یک پاراگراف نیز نتایج را تحلیل کنید و بهترین ترکیب را مشخص کنید.