



پاسخنامه آزمونک شماره ۳

۱- باتوجه به تفاوت‌های بین دو روش بارگیری پویا^۱ و پیوند پویا^۲ به سوالات زیر پاسخ دهید.

الف) در هر کدام از حالت‌های زیر بهتر است که از کدام روش استفاده شود؟ در صورت عدم استفاده از روش انتخابی، چه اتفاقی خواهد افتاد؟ (مقایسه با حالتی که از هیچ‌یک از روش‌ها استفاده نشده است).

- تنها به بخش کوچکی (یک روال^۳ خاص) از کد یک کتابخانه برای اجرای یک پردازش نیاز باشد. (۲.۵ نمره)

بارگذاری پویا (۰.۵ نمره) - با بارگذاری پویا، تنها روتین مورد نیاز بارگذاری می‌شود، که باعث صرفه‌جویی در حافظه می‌شود. اگر از پیوند پویا استفاده کنیم، کل کتابخانه در حافظه بارگذاری می‌شود، حتی اگر فقط به یک بخش کوچک از آن نیاز داشته باشیم. این کار باعث مصرف غیرضروری حافظه می‌شود (۱ نمره). اگر از هیچ کدام از این روش‌ها استفاده نکنیم (پیوند ایستا)، کد کتابخانه مستقیماً در فایل اجرایی برنامه کپی می‌شود. این کار باعث افزایش حجم فایل اجرایی می‌شود (۱ نمره).

- چند پردازش که به صورت هم‌زمان در حال اجرا هستند، برای اجرا به یک کتابخانه نیاز داشته باشند. (۲.۵ نمره)

پیوند پویا (۰.۵ نمره) - با پیوند پویا، کتابخانه در حافظه به اشتراک گذاشته می‌شود، به این معنی که تنها یک نسخه از کتابخانه در حافظه وجود دارد و تمام پردازش‌ها از آن استفاده می‌کنند. این کار باعث صرفه‌جویی قابل توجهی در حافظه می‌شود. اگر از بارگذاری پویا استفاده کنیم، هر پردازش باید نسخه خود را از کتابخانه بارگذاری کند، که منجر به مصرف بیشتر حافظه می‌شود (۱ نمره). اگر از هیچ کدام از این روش‌ها استفاده نکنیم (پیوند ایستا)، هر پردازش یک کپی از کد کتابخانه را در حافظه خواهد داشت، که بدترین حالت از نظر مصرف حافظه است (۱ نمره).

ب) در هر کدام از این روش‌ها، فرایند اتصال کتابخانه‌ها به برنامه اصلی چه زمانی رخ خواهد داد؟ (زمان کامپایل، زمان اجرا یا...) (۱ نمره امتیازی)
هر دو در زمان اجرا انجام می‌شوند (۱ نمره). در پیوند پویا، پیوند نمادین در زمان کامپایل یا پیوند اولیه انجام می‌شود، اما بارگذاری واقعی و اتصال نهایی کتابخانه به فضای آدرس فرآیند در زمان اجرا صورت می‌گیرد.

ج) هر کدام از حالت‌های زیر، راهکار استفاده از کدام‌یک از روش‌های ذکر شده را شرح داده‌اند؟ (۲ نمره)

- در کد برنامه از توابعی برای بارگذاری، به دست آوردن آدرس تابع، و در نهایت آزاد کردن حافظه از کتابخانه استفاده خواهیم کرد.

بارگذاری پویا (۱ نمره)

- هنگام کامپایل یا پیوند برنامه، به کتابخانه مشترک^۴ ارجاع خواهیم داد.

پیوند پویا (۱ نمره)

¹ Dynamic Loading

² Dynamic Linking

³ Routine

⁴ Shared Library

۲- یک سیستم با حافظه مجازی را در نظر بگیرید که از دو لایه جدول صفحه استفاده می‌کند. هر کدام از این جدول‌ها درون یک صفحه قرار می‌گیرند و هر مدخل آن چهار بایتی خواهد بود. هر صفحه نیز چهار کیلوبایت حجم داشته و یک حافظه فیزیکی ۱۶ مگابایتی نیز در اختیار داریم. فرض کنید که هر مدخل در TLB دارای یک بایت فراداده^۵ شامل بیت‌های Valid, Dirty, و دیگر بیت‌ها است. باتوجه به اطلاعات ارائه شده، به سؤالات زیر پاسخ دهید.

الف) در قسمت Offset آدرس‌های این سیستم چند بیت وجود دارد؟ (۱ نمره)

$$Page\ size = 4\ kb = 4 \times 1024 = 4096\ bytes \rightarrow Number\ of\ offset\ bits = \log_2(4096) = \log_2(2^{12}) = 12\ bits$$

ب) تعداد بیت‌های Index جدول صفحه لایه دوم چقدر است؟ (۱ نمره)

$$Number\ of\ entries = \frac{Page\ size}{Entry\ size} = \frac{4096}{4} = 1024 \rightarrow Number\ of\ bits = \log_2(1024) = \log_2(2^{10}) = 10\ bits$$

ج) در صورتی که TLB دارای ۸ مدخل باشد، اندازه آن به بایت را بیابید. (۱ نمره)

$$TLB = VPN + PFN + Metadata = 20 + 12 + 8 = 40\ bits \rightarrow \frac{40}{8} = 5\ bytes$$

$$8\ TLB\ entries \rightarrow 8 \times 5 = 40\ bytes$$

د) فرض کنید اولین کاربری که به این سیستم وارد می‌شود، به آدرس‌های 0x10000000 و 0x10001000 دسترسی پیدا می‌کند و دومین کاربر سیستم فقط به آدرس 0x10000000 دسترسی پیدا می‌کند. در این صورت، چند مدخل در TLB قرار دارد؟ چند صفحه متفاوت در حال حاضر در حافظه قرار دارد؟ (۲ نمره امتیازی)

TLB entries: 2

Distinct pages in memory: 2

۳- یک سیستم با فضای آدرس منطقی شامل ۱۲۸ صفحه و اندازه هر صفحه ۸ کیلوبایت را در نظر بگیرید. این سیستم از حافظه فیزیکی با ۳۲ قاب^۶ استفاده می‌کند. (محاسبات به طور کامل ذکر شوند)

الف) برای آدرس منطقی به چند بیت نیاز است؟ (۱.۵ نمره)

$$128 = 2^7, 8\ kb = 2^3 \times 2^{10} = 2^{13} \rightarrow 13 + 7 = 20\ bit$$

ب) برای آدرس فیزیکی به چند بیت نیاز است؟ (۱.۵ نمره)

$$32 = 2^5 \rightarrow 13 + 5 = 18\ bit$$

⁵ Metadata

⁶ Frame

۴- یک سیستم فرضی با ۴ قاب حافظه فیزیکی و ۸ صفحه حافظه مجازی داریم. دسترسی به صفحات مجازی به ترتیب رشته زیر انجام می‌شود (از چپ به راست):

'ABCADBECFBD AEC'

باتوجه به این اطلاعات، الگوریتم‌های جایگزینی صفحه FIFO، LRU و بهینه^۷ را برای پیاده‌سازی در این سیستم شبیه‌سازی کرده و به سوالات زیر پاسخ دهید. (فرض کنید در الگوریتم بهینه، اگر تفاوتی بین جایگزین کردن چند صفحه نباشد، اولویت جایگزینی با صفحه‌ای است که در شماره قاب کمتر قرار دارد.)

الف) برای هر یک از الگوریتم‌های FIFO، LRU و بهینه، پس از اجرای الگوریتم، کدام صفحات در حافظه فیزیکی باقی خواهند ماند؟ (ترتیب صفحات مهم است) (۳ نمره)

ب) برای هر یک از الگوریتم‌های FIFO، LRU و بهینه، تعداد خطاهای صفحه را محاسبه نمایید. (۳ نمره)

پاسخ دو مورد الف و ب (مواردی که در تصویر بارم ۲ نمره‌ای دارند، در واقع دارای ۱ نمره هستند):

ABCADBECFBD AEC									
LRU:									
A	A	C	C	C	A	A	A	→	11 Page fault (نو)
B	B	B	B	B	B	B	C		
C	E	E	E	D	D	D	D	→	A C D E (ایست)
D	D	D	F	F	F	E	E		(نمره ۳)
FIFO:									
A	E	E	E	E	C			→	9 Page fault (ایز)
B	B	F	F	F	F				
C	C	C	B	B	B			→	C F B A (ایز)
D	D	D	A	A					
Optimal:									
A	E	E	E	C				→	8 Page fault (ایز)
B	B	B	A	A					
C	C	F	F	F				→	C A F D (ایز)
D	D	D	D						

ج) آیا امکان پیاده‌سازی عملی الگوریتم بهینه وجود دارد؟ پاسخ خود را توجیه کنید. (۱.۵ نمره)

خیر (۰.۵). زیرا برای اینکه الگوریتم بهینه تصمیم بگیرد کدام صفحه را جایگزین کند، باید بداند در آینده دقیقاً کدام صفحات مورد استفاده قرار خواهند گرفت. (۱ نمره) و ما نمی‌توانیم رفتار آینده یک فرآیند یا برنامه را به طور دقیق پیش‌بینی کنیم.

د) به طور کلی برای اینکه بتوانیم تخمینی از الگوریتم بهینه داشته باشیم، کدام یک از دو الگوریتم دیگر پیشنهاد داده می‌شود؟ علت انتخاب این الگوریتم چیست؟ آیا در حالت فرضی ارائه شده نیز، این الگوریتم تخمین خوبی ارائه خواهد کرد؟ (با الگوریتم دیگر مقایسه کنید). (۲ نمره)

LRU (۰.۵ نمره) زیرا بر این فرض استوار است که صفحاتی که در گذشته نزدیک استفاده شده‌اند، احتمالاً در آینده نزدیک نیز مورد استفاده قرار خواهند گرفت. در مقابل، صفحاتی که برای مدت طولانی استفاده نشده‌اند، احتمالاً کمتر مورد استفاده قرار می‌گیرند. (۱ نمره) این فرض در بسیاری از برنامه‌ها و الگوهای دسترسی به حافظه صادق است (اصل موضعی). خیر (۰.۵ نمره) در این سناریو تخمین خوبی نسبت به FIFO ارائه نمی‌دهد.

⁷ Optimal

۵- فرض کنید در یک سیستم صفحه‌بندی، زمان دسترسی به TLB^۸ برابر با ۲۰ نانوثانیه و زمان دسترسی به حافظه اصلی برابر با ۱۰۰ نانوثانیه است. اگر نرخ برخورد TLB^۹ برابر با ۹۰٪ باشد، زمان دسترسی مؤثر حافظه^{۱۰} (EMAT) چقدر است؟ (۱ نمره امتیازی)

طول	پایه ^{۱۱}	قطعه
۲۰۰	۱۴۰۰	۰
۴۰۰	۶۳۰۰	۱
۵۰۰	۴۳۰۰	۲
۱۰۰۰	۳۲۰۰	۳
۱۹۰	۴۷۰۰	۴

$$EMAT = (0.9 \times 120ns) + (0.1 \times 220ns) = 130ns$$

۶- فرض کنید یک سیستم از قطعه‌بندی^{۱۲} استفاده می‌کند و دارای جدول قطعه روبرو است. برای هریک از آدرس‌های منطقی زیر، آدرس فیزیکی معادل را محاسبه کرده و یا مشخص کنید که یک خطای قطعه رخ خواهد داد. (۱.۵ نمره) هر مورد ۰.۵ نمره

– (۰, ۱۵۰)

1550

– (۱, ۵۰)

6350

– (۲, ۵۵۰)

$500 < 550 \rightarrow \text{fault}$

⁸ Translation Lookaside Buffer

⁹ TLB Hit Rate

¹⁰ Effective Memory Access Time

¹¹ Base

¹² Segmentation

۷- در صورت استفاده از الگوریتم جایگزینی بهینه در یک سیستم با سه قاب حافظه فیزیکی - که هر سه در ابتدا خالی هستند - و با فرض اینکه طول رشته ارجاع^{۱۳} ۹۹ بوده و از چهار صفحه مجزا (با شماره‌های متفاوت) استفاده شود، حداکثر تعداد خطای صفحه ممکن چقدر است؟ (۱ نمره)

بعد از پر شدن فریم‌ها توسط سه تا از صفحه‌ها، حداکثر با گذراندن هر سه شماره صفحه در رشته ارجاع یک خطای صفحه داریم:

$$\frac{96}{3} + 3 = 35$$

۸- تفاوت اصلی بین یک خطای جزئی صفحه^{۱۴} و یک خطای کلی صفحه^{۱۵} در لینوکس چیست؟ هر کدام از این انواع خطای صفحه^{۱۶} چه زمانی رخ خواهند داد؟ (۲ نمره امتیازی)

یک خطای کلی صفحه زمانی رخ می‌دهد که صفحه مورد نظر فرآیند در حافظه فیزیکی قرار دارد، اما در جدول صفحه فرآیند به درستی نگاشت نشده است. (۱ نمره)

یک خطای جزئی صفحه زمانی رخ می‌دهد که صفحه مورد نظر فرآیند در حافظه فیزیکی وجود نداشته باشد و نیاز به واکنشی از حافظه ثانویه (مانند دیسک) باشد. (۱ نمره)

¹³ Reference String

¹⁴ Minor Page Fault

¹⁵ Major Page Fault

¹⁶ Page Fault