

سیستم‌های عامل

دانشکده مهندسی کامپیوتر

محمدعلی میرزایی و محدثه میربیگی
پاییز ۱۴۰۲



تمرین چهارم

تاریخ تحویل: ۴ دی ۱۴۰۲

اسلایدهای ۹ و ۱۰

پرسش ۱ (۱۰ نمره) به نظر شما همه پیچ‌های حافظه به یک اندازه‌اند؟ این موضوع چه مشکلاتی را ممکن است به وجود آورد؟ تحقیق کنید چه راه‌هایی برای رفع این مشکل وجود دارد.

پرسش ۲ (۲۵ نمره) معماری ساده‌ی زیر را در نظر بگیرید که یک فضای آدرس‌دهی مجازی ۸ بیتی دارد:

map(1)	pg#(3)	offset(4)
--------	--------	-----------

اگر بیت با ارزش بیشتر ۱ باشد، آدرس از طریق MMU مپ می‌شود و در غیر این صورت مپ نشده و آدرس فیزیکی مربوطه شامل ۷ بیت باقی‌مانده‌ی virtual address می‌شود.

همچنین این معماری از یک page table پشتیبانی می‌کند که محتوای آن در زیر آورده شده‌است. هر PTE شامل ۹ بیت می‌شود. یک valid bit و در ادامه ۸ بیت physical page number آمده‌است:

۱	0xFF
۰	0x00
۰	0xA0
۱	0x08
۱	0xB0
۱	0xEE
۱	0xDD
۰	0xFF

(آ) بیش‌ترین اندازه‌ی حافظه‌ی فیزیکی که این معماری پشتیبانی می‌کند چقدر است؟

(ب) بیش‌ترین اندازه‌ی یک single virtual address space چقدر است؟

(ج) اندازه‌ی هر page چقدر است؟

(د) آدرس مجازی 0x63 به چه آدرس فیزیکی مپ می‌شود؟

(ه) چه بازه‌ی از آدرس‌های فیزیکی همواره translation و لیدی از virtual address ها هستند؟

پرسش ۳ (۱۰ نمره)

در یک پردازنده‌ی ۳۲-bit ای معماری x86 با page های 4KB، هر page table یک process شامل چند PTE می‌شود؟ چگونه استفاده‌ی حافظه‌ی پردازنده‌های واقعی اجازه می‌دهد که اندازه‌ی page table کم باشد؟ توضیح دهید چگونه یک آدرس مجازی ۳۲-bit ای که شامل ۱۲ بیت offset و ۱۰ بیت برای level-1 page table و ۱۰ بیت برای level-2 page table است، کار می‌کند؟

پرسش ۴ (۱۰ نمره) در یک حافظه با single-level page table اگر هر دسترسی به حافظه‌ی اصلی به‌طور میانگین ۵۰ نانوثانیه و هر دسترسی به TLB به‌طور میانگین ۱۰ نانوثانیه طول بکشد، به سوالات زیر پاسخ دهید.

• زمان میانگین دسترسی به حافظه برای زمانی که TLB از خود hit rate برابر نیم نشان دهد را با حالتی که TLB نداشته باشیم مقایسه کنید.

• در صورتی که page fault نداشته باشیم، برای رسیدن به زمان میانگین ۶۱ نانوثانیه برای دسترسی به حافظه‌ی اصلی، hit rate باید برابر چند باشد؟

پرسش ۵ (۱۵ نمره) علی می‌خواهد یک حافظه مجازی برای سیستم‌عامل خود پیاده‌سازی کند. او یک Virtual Memory Paging Scheme را به صورت جداول صفحات یک سطحه با ۲۴ بیت آدرس مجازی، ۳۲ بیت فضای آدرس فیزیکی، اندازه صفحه ۶۴ کیلوبایت و ۴ بایت به ازای هر خانه از جدول صفحه طراحی می‌کند. فرض کنید که هر جدول صفحه در یک صفحه جا می‌گیرند و آدرس‌های حافظه Byte-Addressable است.

(آ) کمترین میزان که هر جدول صفحه نیاز دارد؟

(ب) چه تعداد از سطوح از جدول صفحات برای نگاشت کل حافظه آدرس مجازی به آدرس فیزیکی نیاز است؟

(ج) چند بیت از MetaData را میتوان در هر خانه از جدول صفحه جا داد؟

پرسش ۶ (۲۰ نمره) یک سیستم با حافظه‌ی فیزیکی ۴ فریم در نظر بگیرید. این سیستم با حافظه‌ی خالی شروع به کار می‌کند و در ادامه‌ی آن به ترتیب صفحه‌های ۱، ۳، ۵، ۷، ۸، ۱، ۴، ۴، ۳، ۲، ۴، ۵، ۱، ۷، ۱ درخواست می‌شوند. page fault ها را در هر قسمت نشان دهید و تعداد آن‌ها را بشمارید.

(آ) وضعیت حافظه را پس از هر درخواست در صورتی که از الگوریتم FIFO برای جابجایی صفحات استفاده کنیم نشان دهید. علاوه بر این الگوریتمی که اجرا می‌کنید را توضیح دهید.

(ب) وضعیت حافظه با الگوریتم LRU با شانس مجدد را نشان دهید. علاوه بر این الگوریتمی که اجرا می‌کنید را توضیح دهید.

(ج) وضعیت حافظه با الگوریتم LFU را نشان دهید. علاوه بر این الگوریتمی که اجرا می‌کنید را توضیح دهید.

(د) وضعیت حافظه با الگوریتم بهینه را نشان دهید. علاوه بر این الگوریتمی که اجرا می‌کنید را توضیح دهید.

پرسش ۷ (۱۰ نمره) به پرسش‌های زیر پاسخ دهید:

(آ) slab allocator چیست؟ دو مزیت استفاده از این نوع تخصیص دهنده حافظه را بیان کنید.

(ب) به نظر شما به چه علت استفاده از سیاست Local Allocation برای سیستم‌عامل‌های بی‌درنگ (real-time) مناسب است؟ (به وضعیت پیشینی سیستم‌های بی‌درنگ دقت کنید.)