

گزارش کار پروژه ۵ آزمایشگاه سیستم عامل

-محمدسعید صفری

-حبیب قدیمی کلوانق

1) چرا ساختار سلسله مراتبی منجر به کاهش مصرف حافظه می‌گردد؟

اگر از ساختار معمولی برای ذخیره‌ی `page table` ها استفاده کنیم، هر `page table` حافظه‌ای معادل $4 * 1024$ کیلوبایت فضا را اشغال میکند که معادل ۴ مگابایت فضا می‌باشد، حال آنکه در ساختار سلسله مراتبی از یک `page table` کوچک استفاده می‌شود که هر خانه‌ی آن آدرس یک `page table` کوچک دیگر می‌باشد که هر کدام از `page table` های دوم در واقع `page table` حقیقی می‌باشند که آدرس های فیزیکی را دارا می‌باشند، با این روش مقدار حافظه‌ی مصرفی برای ذخیره‌ی آدرس های فیزیکی کاهش پیدا می‌کند.

2) محتوای هر بیت یک مدخل (32 بیتی) در هر سطح چیست؟ چه تفاوتی میان آنها وجود دارد؟

در سطح اول ۲۰ بیت اول آدرس خانه‌ای است که محل آدرس `page table` دوم در آن نگهداری میشود و ۱۲ بیت بعدی دسترسی ها و `flag` ها را مشخص می‌کند. در سطح دوم ۲۰ بیت سمت چپ آدرس فیزیکی یک محل حافظه و ۱۲ بیت بعدی دسترسی ها و `flag` ها را برای آن خانه‌ی حافظه مشخص می‌کند.

2) تابع `(kalloc)` چه نوع حافظه‌ای تخصیص می‌دهد؟ (فیزیکی یا مجازی)

به میزان ۴ کیلوبایت حافظه مجازی اختصاص میدهد. چون کرنل به راحتی با آن کار می‌کند

3) تابع `(mappages)` چه کاربردی دارد؟

این تابع یک آدرس مجازی موجود در یک پردازنده را در سطوح `page table` های آن پردازنده اعمال کرده و به یک آدرس فیزیکی `map` میکند.

4) راجع به تابع `(walkpgdir)` توضیح دهید. این تابع چه عمل سخت‌افزاری را شبیه‌سازی میکند؟

آدرس `PTE` جدول صفحه‌ای را که معادل آدرس مجازی در `page directory` می‌باشد را برمیگرداند.

5) داده‌ساختار `kmem` در فایل `kalloc.c` چه کاربردی دارد؟ `xv6` چگونه صفحات آزاد را ردیابی و نگهداری میکند؟

این داده‌ساختار برای نگهداری `head` لیست `page table` های خالی و در دسترس حافظه مورد استفاده قرار می‌گیرد.

6) در مورد دو فراخوانی سیستمی `(shmget)` و `(shmat)` در سیستم عامل‌های مبتنی بر یونیکس توضیح دهید.

فراخوانی سیستمی `shmget` یک `ID` را به عنوان `shm_ID` گرفته و آن فضای حافظه‌ی اشتراکی را به فضای حافظه‌ی پراسس اضافه میکند، همچنین آرگومان‌هایی را برای تعیین سطح دسترسی حافظه‌ی اشتراکی برای پراسس می‌گیرد، فراخوانی سیستمی `shmat` نیز یک فضای حافظه‌ی اشتراکی را `allocate` کرده و سطح دسترسی های آن را نیز مشخص میکند و `ID` فضای در نظر گرفته شده را برمی‌گرداند.