

به نام خدا

محمد مهدی آقاجانی

تمرین پنجم سیستم عامل

استاد طاهری جوان

تمرین ۱ :

درباره ی ثبات PTBR تحقیق کنید.

آدرس پایه جدول صفحه هر فرآیند در PCB آن قرار میگیرد و ثباتی وجود دارد به نام PTBR که این آدرس در آن قرار میگیرد . لازم به ذکر است که با هر تغییر فرآیند باید مقدار آدرس پایه جدول صفحه جدید از روی PCB آن فرآیند بدست آید و داخل ثبات PTBR پردازنده بارگذاری شود

تمرین ۲ :

الف) درباره ی تکنیک TLB جهت افزایش جهت افزایش سرعت صفحه بندی تحقیق کنید.

ب) فرض کنید در سیستمی زمان دسترسی به حافظه ۶۰ نانو ثانیه و زمان دسترسی به TLB برابر ۵ نانو ثانیه است. اگر احتمال وجود شماره ی صفحه در TLB برابر ۸۰٪ باشد، میانگین زمان دسترسی به حافظه دقیقاً چقدر است؟

الف) در روش صفحه بندی باید به این نکته دقت کنیم که ما برای هر بار دسترسی باید دو بار به RAM مراجعه نماییم و همین امر باعث کندی می شود . به همین دلیل برخی از صفحاتی که پرارجاع هستند را در حافظه انجمنی TLB میریزند که قدرت مقایسه و جست و جو در $O(1)$ را دارد و سرعت دسترسی آن بالاست اما به علت قیمت بالای آن مجبوریم به صورت محدود آن را تهیه نماییم و برخی از صفحات پرارجاع را بر روی آن بگذاریم.

ب)

$$access\ time = \frac{80}{100}(5 + 60) + \frac{20}{100}(5 + 60 + 60) = 77\ ns$$

تمرین ۳:

درباره جدول صفحه معکوس تحقیق کنید(مزایا و معایب آن را بررسی کنید)

در روش استفاده از صفحه بندی یکی از بزرگترین مشکلات این است که تعداد صفحات وقتی بالا میرود حجم جدول صفحه زیاد میشود به گونه ای که حتی شاید در RAM جا نشود!! به همین دلیل جداول صفحه معکوس ارایه شد که در آن به جای اینکه به ازای هر صفحه یک درایه در نظر گرفته شود برای هر قاب یک درایه در نظر گرفته میشود و به همین دلیل سائز جدول بسیار کوچک خواهد بود و همواره ثابت است . اما مشکل عمده آن تبدیل آدرس مجازی به آدرس فیزیکی خواهد بود زیرا برای هر دسترسی باید کل جدول قاب ها پیمایش شود زیرا دیگر شماره صفحه اندیس اصلی نیست بلکه شماره قاب اندیس میباشد و از همین رو نیاز است تا کل جدول گشته شود و سرعت کاهش پیدا میکند مگر اینکه از یک TLB استفاده شود که البته در این صورت هزینه بسیار بالا میرود.

تمرین ۴:

درباره جداول صفحه درهم تحقیق کنید

یک روش متداول برای اداره کردن فضای آدرس بیش از ۳۲ بیت استفاده از جدول صفحه بندی درهم سازی شده است. در این روش یک مقدار درهم سازی به عنوان شماره صفحه مجازی منظور میشود. هرورودی جدول درهم سازی حاوی یک لیست پیوندی از عناصر است که به محل یکسانی درهم سازی میشوند. هر عنصر شامل سه فیلد ۱- شماره صفحه مجازی ۲- مقدار قاب صفحه نگاشت شده ۳- اشاره گری به عنصر بعدی در لیست پیوندی.

شماره صفحه مجازی در آدرس مجازی در جدول درهم سازی درهم سازی میشود. شماره صفحه مجازی با فیلد ۱ در اولین عنصر لیست پیوندی مقایسه میشود. اگر تطبیقی پیدا شد. قاب صفحه متناظر برای ساخت آدرس فیزیکی مطلوب مورد استفاده میشود. اگر تطبیقی پیدا نشود ورودی های بعدی لیست پیوندی جست و جو میشوند تا شماره صفحه مجازی مطلوب پیدا شود. شکل دیگری از این طرح پیشنهاد شده است. که برای فضای آدرس ۶۴ بیتی مناسب است. جدول صفحه ی خوشه ای. مشابه با جدول صفحه درهم سازی هستند. با این تفاوت که هرورودی در جدول درهم سازی به جای یک صفحه به چندین صفحه مراجعه میکند. پس جدول صفحه ای با یک ورودی میتواند نگاشت های مربوط به چندین قاب صفحه فیزیکی را ذخیره نماید. جدول صفحه خوشه ای برای فضای آدرس اسپارس مفید است. در این فضا آدرس ارجاعات به حافظه همجوار نیستند و پراکنده اند.

تمرین ۵:

درباره قطعه بندی همراه با صفحه بندی تحقیق کنید.

در این روش به ازای هر قطعه یک جدول صفحه ساخته میشود . به عبارت دیگر برنامه نویس برنامه را قطعه بندی کرده و سپس سیستم عامل هر قطعه را صفحه بندی میکند . در این فرآیند برای هر فرآیند یک جدول قطعه و برای هر قطعه یک جدول صفحه وجود دارد یعنی هر قطعه از چند صفحه تشکیل میشود که در این حالت آدرس منطقی شامل بخش های شماره قطعه و شماره صفحه و آفست خواهد بود.

تبدیل آدرس نیز به صورت زیر انجام میگیرد :

پس از مراجعه به جدول قطعه برنامه مورد نظر با توجه به محتوای STBR به کمک بخش شماره قطعه در آدرس منطقی ، آدرس پایه متناظر به عنوان آدرس شروع جدول صفحه از جدول قطعه استخراج میشود. سپس این آدرس پایه با مقدار شماره صفحه جمع شده تا شماره صفحه واقعی بدست آید حال به جدول صفحات مراجعه میشود تا شماره قاب صفحه مورد نظر به دست آید

تمرین ۶:

الف) درباره صفحه بندی چند سطحی تحقیق کنید. مزیت عمده آن چیست؟

- ب) سیستمی از جداول صفحه دو سطحی استفاده می کند و دارای آدرس های منطقی 32 بیتی است. اولین هشت بیت آدرس به عنوان ایندکس به جدول سطح اول اشاره دارد و 10 بیت بعدی عناصر جداول سطح دوم را مشخص می کند. به سوالات زیر پاسخ دهید:
1. صفحات در این سیستم چند بیتی هستند؟
 2. فضای آدرس دهی منطقی چند صفحه را شامل می شود؟
 3. جداول سطح اول و سطح دوم هر کدام چند درایه دارند؟

الف) صفحه بندی چند سطحی در واقع روشی است که سعی میکند اندازه جدول صفحه را کاهش دهد به این صورت که بخشی از جدول که مورد نیاز است بر روی RAM قرار گیرد و مابقی بر روی دیسک باشد. مزیت اصلی آن این است که اندازه جدول صفحه را کاهش میدهد ولی عیب آن این است که عملیات ترجمه آدرس کند میشود زیرا با افزایش تعداد سطح تعداد تبدیل ها نیز بیشتر میگردد.

ب)

۱- چون انحراف ۱۴ بیت میشود پس اندازه صفحات 2^{14} بایت میشود

۲- تعداد صفحات $2^{18} = 2^{10} * 2^8$ خواهد بود.

۳- جدول سطح اول 2^8 و جدول سطح دوم 2^{10} درایه خواهد داشت

تمرین ۷:

سیستمی از قطعه بندی ساده استفاده می کند، آدرس فیزیکی هر یک از آدرس های منطقی داده شده را پیدا کنید

الف) ۴۴۰

ب) ۱۹۱

ج) معتبر نیست

د) ۷۰۸

ه) معتبر نیست

تمرین ۸:

چرا سیستم عامل های موبایل مانند iOS و اندروید از تکنیک مبادله (Swapping) (استفاده نمی کنند؟ راهکار جایگزین آنها چیست؟

گرچه اغلب سیستم عامل های مربوط به کامپیوترهای شخصی و سرور ها از نسخه اصلاح شده swapping حمایت می کنند اما سیستم عامل های موبایل اغلب از هیچ شکلی از این تکنیک حمایت نمی کنند. سیستم عامل های موبایل برای حافظه دائم خود عموماً به جای هارد دیسک از flash memory استفاده می کنند. در نتیجه با محدودیت حافظه مواجه خواهند شد و به همین دلیل طراحان سیستم عامل های موبایل از تکنیک swapping اجتناب می کنند. دلیل دیگر اجتناب از این تکنیک ، محدودیت در تعداد نوشتن های روی حافظه flash است قبل از آن که این حافظه غیر قابل اعتماد گردد. هم چنین در این دستگاه ها توان عملیاتی ضعیفی بین حافظه flash و حافظه اصلی وجود دارد.

با توجه به دلایل ذکر شده ، دستگاه هایی مانند iOS وقتی فضای آزاد حافظه از یک حد آستانه کمتر می شود، به جای swapping از اپلیکیشن ها در خواست می کند که به طور داوطلبانه فضای اختصاص داده شده را آزاد کنند. داده های فقط خواندنی (مانند کد ها) از سیستم حذف می شوند و در آینده اگر لازم باشد از حافظه flash مجدداً بارگذاری می شوند. داده ای که که اصلاح شده است (مانند استک) هیچگاه از سیستم حذف نمی شود. با این حال هر اپلیکیشنی که به منظور آزاد کردن فضای کافی در حافظه حذف می شود ، ممکن است توسط سیستم عامل خاتمه یابد. (terminate شود)

سیستم عامل اندروید نیز از تکنیک swapping حمایت نمی کند و استراتژی مشابه iOS اتخاذ می کند. اندروید نیز ممکن است اگر فضای کافی در اختیارش نباشد، یک فرآیند را خاتمه دهد. با این حال قبل از خاتمه دادن به یک فرآیند ، اندروید حالت های کنونی

اپلیکیشن را در حافظه flash ذخیره می کند. در این صورت اپلیکیشن به سرعت می تواند مجددا شروع به کار کند.

به خاطر این محدودیت ها ، توسعه دهندگان سیستم های موبایل باید با دقت حافظه را اختصاص داده و آزاد کنند تا مطمئن شوند اپلیکیشن ها حافظه زیادی مصرف نکنند یا از کمبود حافظه رنج نبرند. هم سیستم عامل اندروید و هم iOS از تکنیک paging حمایت می کنند و توانایی مدیریت حافظه را دارند.