جلسه دوازدهم و چهاردهم

حافظه مجازی:

تکنیکی است که اجازه می دهد بدون این که کل برنامه در حافظه اصلی قرار گیرد اجرا شود.**مزیت مهم** این روش آن است که اجازه می دهد برنامه بزرگتر از حافظه اصلی باشد.علاوه بر این در این روش کاربر حافظه را به صورت آرایه ای فوق العاده بزرگ و یکنواخت می بیند.به عبارتی کاربر(برنامه نویس) خود را درگیر حافظه فیزیکی نمی کند و فرض می کند که بی نهایت حافظه در اختیار دارد و برنامه نویسی ساده می شود.**مزیت دیگر** این است که چون فقط صفحات مورد نیاز از دیسک به حافظه اصلی منتقل می شوند بنابراین زمان کمتری صرف عملI/O می شود از این رو زمان پردازش نیز سریعتر می شود.

جهت پیاده سازی حافظه مجازی دو روش وجود دارد.

- ترکیبی از صفحه بندی و swapping
- ترکیبی از قطعه بندی و swapping

به عبارتی حافظه مجازی به صورت صفحه بندی نیازی پیاده سازی می شود. در این روش مطابق اصل محلی بودن مراجعه به حافظه صفحه یا صفحاتی را که مورد نیاز باشند به حافظه اصلی آورده می شوند و هر وقت به صفحه ای نیاز باشد ولی در حافظه اصلی نباشد آن صفحه مطابق روشی که بعدا خواهیم گفت به حافظه اصلیLoad می شود.

حافظه فیزیکی به یک سری قسمت های مساوی به نام Frame یا قاب تقسیم می شود. و برنامه کاربر نیز به قسمت هائی به اندازه هر Frame به نام page یا صفحه تقسیم می شود.

☐ اگر در ابتدا هیچ صفحه ای در حافظه اصلی نباشد و به محض نیاز، به حافظه اصلی آورده شود به این روش **صفحه بندی نیازی محض** گویند. در این روش هرگز صفحه ای تا وقتی که مورد نیاز نباشد وارد حافظه نمی شود.

کارائی صفحه بندی نیازی

اگر t_m زمان دستیابی به حافظه، p احتمال وقوع خطای صفحه(احتمال این که صفحه در حافظه اصلی نباشد) و t_f زمان لازم برای سرویس دهی به خطای صفحه باشد، آنگاه زمان دسترسی موثر (t_{eff})از فرمول زیر محاسبه می شود.

$$t_{eff} = (1-p)t_m + p \times t_f$$

در اکثر سیستم ها t_m بین 10 تا 200 نانو ثانیه می باشد.

انتساب قاب ها یاframe ها به پردازه ها

- **انتساب مساوی.** در این روش قاب های حافظه اصلی بین پردازه ها به صورت مساوی تقسیم می شود به عنوان مثال اگر حافظه اصلی دارای 12 قاب باشد و 5 پردازه بخواهند به طور هم زمان اجرا شوند به هر کدام از پردازه ها دو قاب تخصیص می یابد و دو قاب باقی مانده به عنوان قاب های بافر در نظر گرفته می شود. پس اگر n قاب داشته باشیم معمولا k تا از این n قاب را به عنوان بافر در نظر گرفته وn-k قاب باقی مانده را مابینm پردازه تقسیم می کنیم.
- **انتساب متناسب.** در این روش تعدا قاب ها متناسب با اندازه پردازه ها به پردازه ها تخصیص می یابد به عنوان مثال اگر حافظه اصلی دارای 12 قاب باشد و دو قاب از آنها به عنوان بافر در نظر گرفته شود در صورتی که اندازه هر قاب 1kباشد و در

سیستم سه پردازه 2k , 3k , 5k به طور همزمان در حال اجرا باشند به پردازه 5k، 5قاب، به پردازه 3k، 3قاب و به پردازه 2k، 2قاب در نظر گرفته می شود.

☐ هیچ کدام از دو روش قبلی الویت را در نظر نمی گیرند می توان کاری کرد که به پردازه ها به نسبت الویتشان قاب تخصیص داد.

در هر مراجعه به حافظه اعمال زیر رخ می دهد

در روش مدیریت حافظه مجازی به آدرس منطقی آدرس مجازی می گویند اگر بیت v یک باشد بدین معناست که page مورد نیاز در حافظه اصلی موجود است اگر صفر باشد دو برداشت از آن وجود دارد.

1- چنینpage ی اصلا وجود ندارد یعنی دسترسی غیر مجاز است به عنوام مثال پردازه شما دارای سه page است ولی شما مراجعه به page چهارم را می دهید که این دسترسی غیر مجاز است.

2- چنین page ی وجود دارد ولی در حافظه اصلی نیست.

در روش حافظه مجازی به طریق صفحه بندی نیازی، در هر مراجعه به حافظه مراحل زیر دنبال می شود.

1- ابتدا به ساختار جدل صفحه مراجعه می شود اگر صفحه مورد درخواست در حافظه اصلی باشد شماره قاب از ساختار جدول صفحه بدست آمده بنابراین آدرس فیزیکی آماده می شود.

2- اگر بیت v=0 باشد و صفحه مراجعه شده در پردازه وجود نداشته باشد، دسترسی غیر مجاز است و پایان پردازه است

3- اگر بیت v=0 باشد ولی صفحه مورد نظر جزء پردازه باشد دراین صورت خطای فقدان صفحه رخ داده و بنابراین

الف. در حافظه اصلی قاب خالی وجود داشته باشد، صفحه مورد نظر از رسانه جانبی به حافظه اصلی منتقل شده و جدول صفحه update می شود.

ب. در حافظه اصلی قاب خالی وجود ندارد از این رو یکی از صفحات موجود در حافظه اصلی مطابق الگوریتم های جایگزینی به رسانه جانبی منتقل شده و صفحه مورد نیاز از رسانه جانبی به حافظه اصلی منتقل می شود.

برای این که انتقال صفحه مورد نیاز از حافظه جانبی به حافظه اصلی منتظر خالی شدن قاب حافظه اصلی نباشد، صفحه مورد نیاز از حافظه جانبی به یکی از قاب های باقر منتقل شده و بنابراین عمل ورودی و خروجی سریعتر انجام میشود. سپس بعد از انتقال قاب شایسته ی خروج از حافظه اصلی به حافظه جانبی،قاب خالی شده به لیست قاب های باقر اضافه می شود.

الگوریتم های جایگزینی صفحه

مساله الگوریتم های جایگزینی صفحه یعنی هنگامی که نیاز است صفحه ای را از حافظه خارج کنیم کدام یک را انتخاب کنیم تا کارائی حداکثر شود. در زیر الگوریتم های متعددی را برای جایگزینی صفحه شرح می دهیم. ولی به طور کلی الگوریتمی بهتر است که تعداد نقض صفحه های(page fault rate) آن کمتر باشد بدیهی است که هر چه قدر تعداد فریم های آزاد بیشتر باشد، تعداد خطا های صفحه ای کاهش می یابد.

1- الگوریتم FIFO : *ساده ترین روش جایگزینی صفحه است. در این روش صفحه ای برای خروج انتخاب می شود که زودتر از بقیه صفحات وارد حافظه اصلی شده باشد*

مثال. مراجعات (از چپ به راست) 4 , 3 , 2 , 1 , 0 , 4 , 1 , 0 , 3 , 2 , 1 , 0 *را با سه قاب آزاد در نظر بگیرید در روش جایگزینی* FIFO *چند نقض صفحه رخ می دهد.*

تقاضا	0	1	2	3	0	1	4	0	1	2	3	4
قاب 1	0	0	0	3	3	3	4	4	4	4	4	4
قاب 2		1	1	1	0	0	0	0	0	2	2	2
قاب 3			2	2	2	1	1	1	1	1	3	3
نقض صفحه	*	*	*	*	*	*	*			*	*	

ملاحظه می شود که در این مثال 9 *بار خطای نقض صفحه رخ می دهد.*

ناهنجاری بلیدی: *اگر در الگوریتم جایگزینی با افزایش تعدادقاب ها*(Frame) *تعداد خطای صفحه افزایش یابد گوئیم آن الگوریتم دارای ناهنجاری بلیدی است که الگوریتم* FIFO *یکی از آنها می باشد.*

ادامه دارد.....