



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)



دانشکده مهندسی کامپیوتر

به نام خدا

پاسخ تمرین سری اول درس سیستم های عامل

ترم بهار 1404

استاد درس: دکتر زرندی

سوال اول)

الف) در وقفه اجرای دستورالعمل‌های جاری پردازنده موقتاً متوقف شده و دستورات سرویس‌دهی دیگری اجراء می‌گردد و سپس ، عمل پردازش دوباره به برنامه وقفه‌داده‌شده باز می‌گردد.

وقفه مکانیزیمی است که طی آن سخت‌افزارها می‌توانند وضعیت یا نیازمندی‌های خودشان را به اطلاع پردازشگر برسانند. از راهکارهای وقفه افزایش کارائی پردازنده می باشد .

ب)

- برنامه (Program):

زمانی ایجاد می‌شود که شرایطی در اجرای یک دستورالعمل رخ دهد، مانند سرریز عددی، تقسیم بر صفر، تلاش برای اجرای یک دستورالعمل غیرمجاز ماشین، و ارجاع به خارج از محدوده حافظه مجاز کاربر.

- زمان‌سنج (Timer):

توسط یک زمان‌سنج درون پردازنده ایجاد می‌شود. این امکان را به سیستم‌عامل می‌دهد که به طور منظم عملکردهای خاصی را انجام دهد.

- ورودی/خروجی (I/O):

توسط یک کنترل‌کننده ورودی/خروجی ایجاد می‌شود تا تکمیل عادی یک عملیات را اعلام کند یا برای اعلام انواع مختلفی از شرایط خطا استفاده شود.

- خرابی سخت‌افزار (Hardware failure):

به دلیل خرابی ایجاد می‌شود، مانند قطع برق یا خطای توازن حافظه (memory parity error).

ج) مربوط به اجرای ISR می‌باشد.

سوال دوم)

1. ممتاز نیست. خواندن ساعت توسط برنامه‌های کاربر تغییری در سیستم ایجاد نمی‌کند.
2. ممتاز نیست. تله یک وقفه‌ی نرم‌افزاری است که توسط خطا در برنامه‌های کاربر یا درخواست کاربر ایجاد می‌شود و باعث انتقال از حالت کاربر به هسته می‌شود.
3. ممتاز است. سیستم‌عامل با استفاده از وقفه‌ها کارهای مهمی از جمله مدیریت دستگاه‌های IO را انجام می‌دهد. پس در صورتی که برنامه‌های کاربر امکان منع وقفه‌ها را داشته باشند، تغییرات نامطلوبی در سیستم ایجاد می‌کنند.
4. ممتاز است. تغییر به حالت هسته باید تنها از طریق رویه‌های مشخص سیستم‌عامل انجام شود. اگر برنامه‌ی کاربر بتواند این دستورعمل را اجرا کند، می‌تواند در هر زمان به حالت هسته وارد شود و به کل سیستم دسترسی داشته باشد.
5. ممتاز است. ممکن است به طور همزمان چندین درخواست برای خواندن از یا نوشتن به دستگاه IO از طرف چند برنامه‌ی کاربر وجود داشته باشد. پس لازم است که این درخواست‌ها توسط سیستم‌عامل مدیریت شوند.
6. ممتاز نیست. نوشتن در PC می‌تواند توسط برنامه‌های کاربر در طی اجرای دستورعمل‌های برنامه مانند پرش به یک آدرس خاص، اجرای تابع یا برگشت از تابع انجام شود. باید از دسترسی برنامه به آدرس‌های غیرمجاز مثلاً آدرس‌های مربوط به برنامه‌های دیگر جلوگیری کرد ولی نگه داشتن PC در محدوده‌ی امن، سخت‌افزاری انجام می‌شود.
7. ممتاز نیست. سیستم‌عامل مسئول زمانبندی پردازش‌ها است اما پردازنده خودش اولویت‌های داخلی را تغییر می‌دهد (مثلاً بر اساس میزان بار پردازشی). به طور کلی می‌توان گفت که پردازنده‌ها مکانیزم‌های داخلی برای تغییر اولویت هسته دارند.

سوال سوم)

الف) با افزایش تعداد پردازنده‌ها، انتظار داریم که کار بیشتری در زمان کمتری انجام شود. اما افزایش سرعت با N پردازنده برابر با N نیست؛ بلکه کمتر از N است. زمانی که چند پردازنده برای انجام یک وظیفه با هم همکاری می‌کنند، مقداری بار اضافی (Overhead) برای هماهنگ نگه داشتن همه اجزا و کارکرد صحیح آن‌ها ایجاد می‌شود. این بار اضافی، به همراه رقابت برای منابع مشترک، باعث کاهش سود حاصل از افزودن پردازنده‌های بیشتر می‌شود.

ب) دسترسی غیر یکنواخت به حافظه: در این رویکرد، هر پردازنده (یا گروهی از پردازنده‌ها) حافظه محلی خود را دارد که از طریق یک باس محلی سریع و کوچک به آن دسترسی پیدا می‌کند. پردازنده‌ها توسط سیستم ارتباطی مشترک (Shared System Interconnect) به هم متصل شده‌اند تا همه پردازنده‌ها یک فضای آدرس فیزیکی مشترک داشته باشند. این رویکرد که به آن دسترسی غیر یکنواخت به حافظه (NUMA) می‌گویند، به پردازنده‌ها اجازه می‌دهد که به حافظه محلی خود سریع‌تر دسترسی داشته باشند و رقابت بر سر اتصال سیستم وجود نداشته باشد.

دسترسی یکنواخت به حافظه: دسترسی به تمام حافظه اصلی توسط تمامی پردازنده‌ها، بدون تفاوت عملکرد بر اساس مکان پردازنده یا حافظه.

مزیت سیستم غیر یکنواخت این است که وقتی یک پردازنده به حافظه محلی خود دسترسی پیدا می‌کند، علاوه بر سریع بودن، هیچ رقابتی بر سر اتصال سیستم وجود ندارد. بنابراین، سیستم‌های NUMA می‌توانند با افزودن پردازنده‌های بیشتر به طور مؤثرتری مقیاس‌پذیر شوند.

ج) زمانی که یک پردازنده باید به حافظه پردازنده‌های دیگر از طریق سیستم ارتباطی مشترک (Shared System Interconnect) دسترسی پیدا کند که ممکن است منجر به کاهش عملکرد شود. به عنوان مثال پردازنده CPU0 نمی‌تواند به حافظه محلی پردازنده CPU3 به سرعتی که به حافظه محلی خودش دسترسی دارد، دست یابد که این مساله می‌تواند باعث کاهش عملکرد شود. سیستم‌های عامل (نرم‌افزاری) می‌توانند با زمان‌بندی دقیق پردازنده‌ها و مدیریت حافظه این مشکل را کاهش دهند.

سوال چهارم)

بنابر اطلاعات داده‌شده در صورت سوال نرخ انتقال اطلاعات بین حافظه اصلی و حافظه مجازی 50 MB/Sec است. با توجه به اینکه اندازه هر فرآیند 10 MB است. بنابراین انتقال هر فرآیند بین حافظه اصلی و حافظه مجازی به $200 \text{ msec} = \frac{10 \text{ MB}}{50 \text{ MB/Sec}}$ زمان نیاز دارد. حال باتوجه به اینکه cpu time هر فرآیند نیز 200 msec است و بنابه فرض سوال فرآیندها امکان اجرای هم‌زمان با DMA را دارند. بنابراین زمانی که یک فرآیند در حال سپری کردن زمان پردازش خودش است، DMA فرآیند دیگری را از حافظه مجازی به اصلی انتقال داده و از آنجایی که این زمان انتقال با زمان CPU فرآیند در حال اجرا برابر است، CPU به محض اتمام یک فرآیند، فرآیند دیگری برای اجرا در حافظه اصلی خواهد داشت. که در این صورت CPU هیچگاه در صورت وجود فرآیندی برای اجرا بیکار نمانده. بنابراین بهره‌وری آن 100٪ خواهد بود.