

موسسه بابان

انتشارات بابان و انتشارات راهیان ارشد

درس و کنکور ارشد

سیستم عامل

زمان بندی نخها

(به روش دوسطحی در سیستم های بی درنگ نهفته)

ویژه ی داوطلبان کنکور کارشناسی ارشد مهندسی کامپیوتر و IT

براساس کتب مرجع

آبراهام سیلبرشاتز، ویلیام استالینگز و اندور اس تنن بام

ارسطو خلیلی فر

تست‌های فصل سوم

۹۷- در سیستم‌های تعبیه شده بی‌درنگ سخت که پاسخ در زمانی مشخص باید تضمین شود، کدام روش نگاشت ریسمان‌های کاربر به ریسمان‌های سیستمی، مناسب است؟
(مهندسی II - دولتی ۹۷)

(۱) یک به یک

(۲) چند به یک

(۳) چند به چند

(۴) دوسطحی

پاسخ‌های فصل چهارم

۹۷- گزینه (۴) صحیح است.

به طور کلی مدیریت و زمان‌بندی نخ‌ها به سه روش زیر انجام می‌گردد:

۱- روش سطح کاربر یا مدل چند به یک (many to one)

در این روش فقط زمان‌بند پردازنده و زمان‌بند چند نخ‌ی در سطح کاربر وجود دارد و زمان‌بند چند نخ‌ی در سطح هسته در این روش مورد استفاده قرار نمی‌گیرد. در واقع هسته سیستم عامل فقط فرآیندها را می‌شناسد و هیچ اطلاعاتی از نخ‌ها ندارد. در واقع اولویت‌بندی نخ‌ها، مدیریت نخ‌ها و زمان‌بند چندنخی در سطح کاربر و توسط یک بسته نرم‌افزاری انجام می‌گردد. بدین معنی که نخ‌ها را برنامه‌نویس مشخص می‌کند و مدیریت آن‌ها را نیز بر عهده می‌گیرد. بنابراین زمان‌بند پردازنده، براساس الگوریتم مشخصی مثلاً نوبت چرخشی پردازنده را در اختیار یکی از فرآیندهای آماده قرار می‌دهد. سپس زمان‌بند چند نخ‌ی در سطح کاربر، متناسب با کاربردی که در آن فرآیند به کار گرفته می‌شود، الگوریتم زمان‌بند را انتخاب کرده و تصمیم می‌گیرد که پردازنده در اختیار کدام یک از نخ‌های آماده در فرآیند موردنظر قرار گیرد و تا زمانی که پردازنده در تملک فرآیند باشد و یا تا قبل از پایان برش زمانی مربوط به فرآیند، نخ‌های یک فرآیند از پردازنده بهره‌مند می‌شوند و به محض مسدود شدن یک نخ، و یا پایان برش زمانی یک فرآیند، یا اتمام فرآیند، پردازنده به فرآیند بعدی تعلق می‌گیرد.

توجه: در این روش نخ ماهیت منطقی دارد و از دید کاربر فقط وجود دارد، در واقع از نظر سیستم عامل ماهیت فیزیکی ندارد، بنابراین نخ کاربر در این روش همانند یک تابع در فرآیند می‌باشد که از رجیستر و پشته مختص به خود نیز بهره‌مند نمی‌باشد.

توجه: مدل غیر کامپیوتری این روش نیز وجود دارد، مانند حالتی که درآمدهای دولت حاصل از منابع کشور، بین پدران خانواده‌ها تقسیم گردد و این پدران خانواده‌ها باشند که تصمیم بگیرند به هر عضو خانواده چه مقدار نقدینگی تعلق بگیرد. در این روش فقط پدران شماره حساب مختص به خود را دارند. اما اگر یکی از اعضای خانواده خطایی انجام دهد و محکوم گردد، آنگاه تمام اعضای خانواده برای مدتی از خدمات دولت محکوم می‌گردند، زیرا در این مدل، دولت فقط پدران خانواده را می‌شناسد و از اعضای خانواده اطلاعی ندارد. بنابراین حساب پدر خانواده برای مدتی مسدود می‌گردد.

توجه: عمل تعویض متن مابین نخ‌های یک فرآیند کاربر، کاملاً در سطح کاربر و با سربار بسیار ناچیز (در حد فراخوانی نخ بعدی) و بدون تغییر حالت پردازنده به مد هسته پردازنده، توسط زمان‌بند چندنخی در سطح کاربر (برنامه‌های کاربر) انجام می‌گردد. اما عمل تعویض متن مابین فرآیندها (فرآیندهای کاربر یا فرآیندهای سیستم عامل) در سطح هسته سیستم عامل و در مد هسته

پردازنده، توسط زمان‌بند پردازنده برای انتخاب یک فرآیند جدید بر اساس یک الگوریتم خاص انجام می‌گردد.

توجه: به دلیل آنکه نخ‌های یک فرآیند کاربر، تماماً در فضای کاربر مدیریت می‌شوند، اگر نخ موجود در یک فرآیند کاربر، یک فراخوان سیستمی مسدودکننده را اجرا نماید، هسته سیستم عامل نه تنها آن نخ، بلکه کل فرآیند کاربر را که شامل تمام نخ‌های دیگر می‌باشد، مسدود می‌کند، زیرا هسته سیستم عامل خبری از نخ‌های داخل فرآیند کاربر ندارد. در واقع در این روش هسته سیستم عامل نخ‌ها را همانند توابع داخل یک فرآیند می‌بیند، یعنی هسته سیستم عامل، یک فرآیند چند نخی سطح کاربر را، مانند یک فرآیند تک نخی اما دارای چند تابع مختلف می‌بیند!

توجه: این راه‌کار، منجر به عدم امکان هم‌روندی (در سیستم‌های تک‌پردازنده‌ای) نخ‌های داخل یک فرآیند کاربر در حالت انسداد یک نخ داخل یک فرآیند کاربر می‌گردد. البته اگر نخی داخل یک فرآیند کاربر مسدود نگردد، امکان هم‌روندی میان نخ‌های داخل فرآیند کاربر برقرار است. مانند یک تیم فوتبال ۱۱ نفره که اگر بازیکنی مرتکب خطا گردد، از آن‌جا که داور فقط نام تیم را می‌شناسد و نه تک‌تک بازیکنان تیم را، آن‌گاه کل تیم را جریمه، اخراج و مسدود می‌کند. اما اگر هیچ‌یک از بازیکنان تیم مرتکب خطایی نگردد، واضح است که هم‌روندی برقرار است.

توجه: فرض کنید یک کیک داریم که آن را به چهار قسمت مساوی تقسیم کرده‌ایم، همچنین فرض کنید خوردن هر بخش کیک یک ساعت زمان بخواهد، اگر یک نفر بخواهد تمام این کیک را بخورد، پس ۴ ساعت طول می‌کشد اگر ۴ نفر بخواهند تمام این کیک را بخورند و به هر نفر یک بخش کیک داده شود، آنگاه خوردن تمام کیک به‌طور موازی ۱ ساعت طول خواهد کشید. در روش سطح کاربر، یک فرآیند چند نخی کاربر نمی‌تواند از امتیازات چند پردازنده‌ای بهره‌بردار، زیرا در روش سطح کاربر، هسته سیستم عامل در هر لحظه فقط یک پردازنده را در اختیار نخ‌های یک فرآیند کاربر قرار می‌دهد. حتی اگر چند پردازنده موجود باشد. یعنی در این روش خوردن کیک بخش‌بندی شده به صورت چند نفری امکان‌پذیر نیست. بنابراین در این روش امکان پردازش موازی نخ‌های داخل یک فرآیند کاربر وجود ندارد.

توجه: نرم‌افزارهای POSIX P-threads و Mach C-thread به عنوان یک بسته نرم‌افزاری، می‌توانند جهت مدیریت نخ‌ها و زمان‌بند چند نخی در سطح کاربر مورد استفاده قرار گیرند.

توجه: در این روش تخصیص منابع و زمان‌بندی پردازنده، بر روی فرآیندها انجام می‌شود. همچنین زمان‌بندی نخ‌های سطح کاربر، بر عهده زمان‌بند چندنخی سطح کاربر خواهد بود.

توجه: سیستم عامل سولاریس، مدل چند به یک را پیاده‌سازی می‌کند.

۲- روش سطح هسته یا مدل یک به یک (one to one)

در این روش فقط زمان‌بند پردازنده و زمان‌بند چندنخی در سطح هسته وجود دارد و زمان‌بند

چند نخ‌ی در سطح کاربر در این روش مورد استفاده قرار نمی‌گیرد. در واقع هسته سیستم عامل نه تنها فرآیندها را می‌شناسد بلکه از وجود نخ‌های داخل یک فرآیند نیز در صورت وجود نخ آگاه است. در واقع مدیریت نخ‌ها و زمان‌بند چندنخی (نخ‌های فرآیندهای کاربر و نخ‌های فرآیندهای سیستم عامل) در سطح هسته، توسط هسته سیستم عامل انجام می‌گردد و کاربر هیچ دیدی از این کار ندارد. انگار که اجتماع تمام نخ‌های فرآیندهای سیستم عامل و کاربر را در نظر بگیرید، حال بر روی تک نخ‌ها بر اساس یک الگوریتم خاص حرکت کنید. بنابراین زمان‌بند چند نخ‌ی در سطح هسته، بر اساس الگوریتم مشخصی مثلاً نوبت چرخشی پردازنده را در اختیار یکی از نخ‌های آماده قرار می‌دهد. توجه کنید در این روش پردازنده دیگر در تملک فرآیند نیست بلکه در تملک نخ‌ها است. در واقع تا زمانی که پردازنده در تملک یک نخ باشد و تا قبل از مسدود شدن نخ، و یا تمام شدن نخ و یا پایان برش زمانی مربوط به نخ، نخ می‌تواند از پردازنده بهره‌برد و به محض مسدود شدن یک نخ، یا پایان برشی زمان مربوط به نخ یا اتمام نخ، پردازنده می‌تواند به نخ بعدی که ممکن است، نخ بعدی، نخ هم‌خانواده با نخ قبلی در یک فرآیند باشد، یا نخ‌ی در یک فرآیند دیگر باشد، تعلق بگیرد. در واقع زمان‌بند چند نخ‌ی در سطح هسته سیستم عامل تعیین می‌کند که نخ بعدی که باید شروع به کار کند متعلق به همان فرآیند باشد و یا از یک فرآیند دیگر انتخاب شود.

توجه: برای انجام زمان‌بندی چند نخ‌ی، هسته سیستم عامل باید علاوه بر جدول فرآیندها، یک جدول نخ (شبه جدول فرآیند) داشته باشد که اطلاعات تمامی نخ‌های موجود در سیستم را نگهداری کند. مجدداً تأکید می‌کنیم که در این حالت هر نخ TCB خاص خود را دارد. به عبارت دیگر هر نخ رجیستر و پشته مختص به خود را دارد.

توجه: در این روش نخ ماهیت فیزیکی دارد، و از دید کاربر و سیستم عامل وجود دارد، بنابراین در این روش، هر نخ، رجیستر و پشته مختص به خود را دارد، به عبارت دیگر هر نخ TCB مختص به خود را دارد.

توجه: عملیات مرتبط با نخ‌های سطح کاربر، مانند ایجاد (بارگذاری TCB مختص به نخ) و پایان دادن (ذخیره‌سازی TCB مختص به نخ) بر عهده هسته سیستم عامل است و توسط زمان‌بند چند نخ‌ی در سطح هسته سیستم عامل انجام می‌گردد.

توجه: مدل غیر کامپیوتری این روش نیز وجود دارد، مانند حالتی که درآمدهای دولت حاصل از منابع کشور، بدون اعمال هیچ‌گونه اولویت‌بندی بین تک تک اعضای یک کشور تقسیم گردد (حال این اعضا شهروند عام باشد یا خاص) در این روش هر یک از اعضای کشور شماره حساب مختص به خود را دارند. اما اگر یکی از افراد کشور خطایی انجام دهد و محکوم گردد، آنگاه فقط همان فرد برای مدتی از خدمات دولت محروم می‌گردد و دولت به بقیه اعضا خانواده آن فرد همچنان خدمات ارائه می‌دهد. زیرا دولت از مشخصات تک تک اعضا جامعه آگاه است.

توجه: عمل تعویض متن، مابین نخ‌های یک فرآیند (فرآیندهای کاربر یا فرآیندهای سیستم

عامل) در سطح هسته سیستم عامل، در مد هسته پردازنده توسط زمان‌بند چندنخی در سطح هسته سیستم عامل انجام می‌گردد و عمل تعویض متن مابین فرآیندها (فرآیندهای کاربر یا فرآیندهای سیستم عامل) در سطح هسته سیستم عامل و در مد هسته پردازنده توسط زمان‌بند پردازنده انجام می‌گردد.

توجه: به دلیل آنکه نخ‌های یک فرآیند (فرآیندهای کاربر یا فرآیندهای سیستم عامل)، تماماً در فضای هسته سیستم عامل مدیریت می‌شوند و هسته سیستم عامل از وجود نخ‌های یک فرآیند آگاه است، اگر نخ موجود در یک فرآیند، یک فراخوان سیستمی مسدود کننده را اجرا نماید، هسته سیستم عامل فقط آن نخ مربوطه را مسدود می‌کند و نخ‌های دیگر هم‌خانواده با آن نخ مسدود شده، همچنان می‌توانند از پردازنده بهره ببرند.

توجه: این راه کار، منجر به امکان هم‌روندی (در سیستم‌های تک پردازنده‌ای) و امکان توازی (در سیستم‌های چندپردازنده‌ای) میان فرآیندهای مختلف و یا میان نخ‌های داخل یک فرآیند می‌شود. البته اگر نخ مسدود نگردد، درجه هم‌روندی و توازی بالاتر هم خواهد رفت، زیرا در اینصورت همه نخ‌ها به طور هم‌روند یا موازی در حال حرکت هستند. مانند یک تیم فوتبال ۱۱ نفره که اگر بازیکنی مرتکب خطا گردد، هم‌روندی یا توازی همچنان برقرار است، چون داور تک تک بازیکنان را می‌شناسد و فقط بازیکن خاطی را مسدود، جریمه و اخراج می‌کند و بقیه بازیکنان تیم به بازی خود ادامه می‌دهند، اما اگر هیچ یک از بازیکنان تیم مرتکب خطا نگردند، واضح است که هم‌روندی و توازی بالاتر هم خواهد بود.

توجه: مثال کیک مطرح شده را مجدداً به یاد آورید، در روش سطح هسته، یک فرآیند چندنخی (فرآیندهای کاربر یا فرآیندهای سیستم عامل) می‌تواند از امتیازات چندپردازنده‌ای بهره برد. در روش سطح هسته، هسته سیستم عامل می‌تواند در هر لحظه چندین پردازنده را در اختیار نخ‌های یک فرآیند قرار دهد. یعنی در این روش خوردن کیک بخش‌بندی شده به صورت چند نفری امکان‌پذیر است. بنابراین در این روش امکان پردازش موازی نخ‌های داخل یک فرآیند وجود دارد.

توجه: در این روش، منابع به فرآیندها اختصاص می‌یابد ولی زمان‌بندی پردازنده، بر روی نخ-ها انجام می‌گیرد و زمان‌بندی نخ‌های سطح کاربر و نخ‌های سطح هسته، برعهده زمان‌بند چندنخی سطح هسته می‌باشد.

توجه: سیستم عامل لینوکس، خانواده سیستم عامل ویندوز و سولاریس ۹ مدل یک به یک را پیاده‌سازی می‌کنند.

۳- روش ترکیبی (سطح کاربر و هسته) یا مدل چند به چند (many to many)

این روش از اجتماع دو روش سطح کاربر و هسته ابداع گردیده است. در این روش علاوه بر زمان‌بند پردازنده و زمان‌بند چندنخی در سطح هسته سیستم عامل، زمان‌بند چند نخی نیز در سطح

کاربر برای زمان‌بندی و اولویت‌دهی نخ‌های فرآیند کاربر وجود دارد. در واقع زمان‌بندی نخ‌های فرآیندهای سیستم عامل، توسط هسته سیستم عامل و زمان‌بندی نخ‌های فرآیندهای کاربر، توسط برنامه کاربر انجام می‌گردد، که این امر منجر به اولویت‌بندی نخ‌های فرآیندهای کاربر می‌گردد. در این روش، زمان‌بند چند نخ در سطح کاربر، نخ‌های کاندید خود را از میان نخ‌های متعدد در فرآیندهای مختلف کاربر براساس یک الگوریتم خاص انتخاب و تحویل یک زمان‌بند چند نخ در سطح هسته می‌دهد. در واقع انتخاب نخ‌های کاندید (زمان‌بندی) در فرآیندهای کاربر به خود کاربر واگذار شده است که همانطور که گفتیم منجر به اولویت‌بندی نخ‌های فرآیندهای کاربر می‌گردد. حال اجتماع حاصل از نخ‌های کاندید فرآیندهای کاربر و نخ‌های فرآیندهای سیستم عامل توسط زمان‌بند چندنخی هسته سیستم عامل براساس یک الگوریتم خاص زمان‌بندی می‌شود.

از توضیحات فوق این مفهوم برداشت می‌شود که هسته سیستم عامل باید این قابلیت را داشته باشد که کاربر بتواند نخ‌های فرآیندهای سطح خود را به هسته سیستم عامل معرفی کند. بنابراین در این روش هسته سیستم عامل نه تنها فرآیندهای کاربر را می‌شناسد، بلکه از وجود نخ‌های کاربر داخل فرآیندهای کاربر نیز در صورت وجود نخ آگاه است.

توجه: برای معرفی نخ‌های کاندید فرآیندهای سطح کاربر به زمان‌بند چندنخی در سطح هسته سیستم عامل، علاوه بر جایگاه‌های مخصوص نخ‌های فرآیندهای سیستم عامل در زمان‌بند چندنخی در سطح هسته سیستم عامل، تعدادی جایگاه، ویژه نخ‌های سطح کاربر نیز، در زمان‌بند چندنخی در سطح هسته سیستم عامل در نظر گرفته شده است، به این جایگاه **ویژه که محیط اجرای نخ** نیز نامیده می‌شود، **LWP** گفته می‌شود. در واقع هر نخ کاندید انتخاب شده توسط زمان‌بند چندنخی در سطح کاربر، پس از معرفی به زمان‌بند چندنخی در سطح هسته، در یکی از جایگاه‌های ویژه که همان **LWP** است، جهت زمان‌بندی توسط زمان‌بند چندنخی در سطح هسته، قرار می‌گیرد.

توجه: **LWP** سرواژه‌ی عبارت **Light Weight Process** و به معنی فرآیند سبک وزن است. **توجه:** هنگامی که یک نخ به زمان‌بند چندنخی در سطح هسته معرفی می‌گردد و در ادامه در یک **LWP** جهت زمان‌بندی توسط زمان‌بند چندنخی در سطح هسته، قرار می‌گیرد، در طول حیات خود ممکن است، در **LWP** های متفاوتی بخش‌هایی از اجرای خود را طی کند، مثلاً یک نخ در صورت رسیدن به عملیات ورودی و خروجی، جایگاه خود یعنی **LWP** را واگذار می‌کند و در اجرای بعدی پس از پایان عملیات ورودی و خروجی ممکن است به یک **LWP** دیگر منتسب شود. توجه کنید که **LWP محیط اجرای نخ** می‌باشد.

توجه: برای انجام زمان‌بندی چندنخی، هسته سیستم عامل باید علاوه بر جدول فرآیندها، یک جدول نخ (شبه جدول فرآیند) داشته باشد که اطلاعات تمامی نخ‌های موجود در سیستم را نگهداری کند.

توجه: در این روش نخ ماهیت فیزیکی دارد و از دید کاربر و سیستم عامل وجود دارد،

بنابراین در این روش، هر نخ، رجیستر و پشته مختص به خود را دارد، یه عبارت دیگر هر نخ TCB مختص به خود را دارد.

توجه: عملیات مرتبط با نخ‌های سطح کاربر، مانند ایجاد (بارگذاری TCB مختص به نخ) و پایان دادن (ذخیره‌سازی TCB مختص به نخ) بر عهده هسته سیستم عامل است و توسط زمان‌بند چندنخی سطح هسته انجام می‌گردد.

توجه: مدل غیرکامپیوتری این روش نیز وجود دارد، مانند حالتی که درآمدهای دولت حاصل از منابع کشور، با اعمال نوعی اولویت‌بندی بین برخی از اعضاء یک کشور تقسیم گردد (حال این اعضاء شهروند عام باشند یا خاص) در این روش هریک از اعضاء اولویت‌دار و منتخب کشور شماره حساب مختص به خود را دارند. اما اگر یکی از اعضاء خانواده خطایی انجام دهد و محکوم گردد، آنگاه فقط همان فرد برای مدتی از خدمات دولت محروم می‌گردد و دولت به بقیه اعضاء خانواده آن فرد، همچنان خدمات ارائه می‌دهد، زیرا دولت از مشخصات تک تک اعضاء آن خانواده آگاه است.

توجه: عمل تعویض متن مابین نخ‌های یک فرآیند (فرآیندهای کاربر یا فرآیندهای سیستم عامل) در سطح هسته سیستم عامل، در مد هسته پردازنده توسط زمان‌بند چندنخی در سطح هسته سیستم عامل انجام می‌گردد و عمل تعویض متن مابین فرآیندها (فرآیندهای کاربر یا فرآیندهای سیستم عامل) در سطح هسته سیستم عامل و در مد هسته پردازنده توسط زمان‌بند پردازنده انجام می‌گردد.

توجه: به دلیل آنکه نخ‌های کاندید فرآیندهای سطح کاربر، به هسته سیستم عامل معرفی می‌گردند و هسته سیستم عامل از وجود نخ‌های کاندید فرآیندهای سطح کاربر آگاه است، اگر یک نخ کاندید موجود در یک فرآیند کاربر، یک فراخوان سیستمی مسدودکننده را اجرا نماید، هسته سیستم عامل فقط آن نخ مربوطه را مسدود می‌کند و نخ‌های دیگر هم‌خانواده با آن نخ مسدود شده، همچنان می‌تواند از پردازنده بهره ببرند. این راه‌کار، منجر به امکان هم‌روندی (در سیستم‌های تک‌پردازنده‌ای) و امکان توازی (در سیستم‌های چندپردازنده‌ای) میان فرآیندهای مختلف کاربر و یا میان نخ‌های داخل یک فرآیند کاربر می‌شود. البته اگر نخی مسدود نگردد، درجه هم‌روندی و توازی بالاتر هم خواهد رفت، زیرا در اینصورت همه نخ‌ها به طور هم‌روند یا موازی در حال حرکت هستند.

مانند یک تیم فوتبال ۱۱ نفره که اگر بازیکنی مرتکب خطا گردد، هم‌روندی یا توازن همچنان برقرار است، چون داور تک تک بازیکنان را می‌شناسد و فقط بازیکن خاطی را مسدود، جریمه و اخراج می‌کند و بقیه بازیکنان تیم به بازی خود ادامه می‌دهند، اما اگر هیچ‌یک از بازیکنان تیم مرتکب خطا نگردد، واضح است هم‌روندی و توازی بالاتر هم خواهد رفت.

توجه: همانند نخ‌های سطح کاربر، هم‌روندی و توازی برای نخ‌های سطح هسته سیستم عامل

نیز، در حالت انسداد یک نخ یا عدم انسداد یک نخ، برقرار است.

توجه: مثال کیک مطرح شده را مجدداً به یاد آورید، در روش ترکیبی، یک فرآیند چندنخی (فرآیندهای کاربر یا فرآیندهای سیستم عامل) می‌تواند از امتیازات چندپردازنده‌ای بهره‌برد. در روش ترکیبی، هسته سیستم عامل می‌تواند در هر لحظه چندین پردازنده را در اختیار نخ‌های یک فرآیند قرار دهد. یعنی در این روش خوردن کیک بخش‌بندی شده به صورت چند نفری امکان پذیر است. بنابراین در این روش امکان پردازش موازی نخ‌های داخل یک فرآیند وجود دارد.

توجه: در این روش نیز، منابع به فرآیندها اختصاص می‌یابد ولی زمان‌بندی پردازنده، بر روی نخ‌ها، انجام می‌گیرد. اما زمان‌بندی نخ‌های سطح کاربر، بر عهده زمان‌بند چندنخی سطح کاربر و زمان‌بندی نخ‌های سطح هسته، بر عهده زمان‌بند چندنخی سطح هسته خواهد بود.

در نرم‌افزارهای بی‌درنگ باید خروجی و پاسخ نهایی در یک زمان مشخص و از پیش تعیین شده حاصل شود. در این نرم‌افزارها، زمان نقشی کلیدی ایفا می‌کند و زمان پاسخ باید به موقع و تضمین شده باشد. نرم‌افزارهای بی‌درنگ معمولاً به عنوان یک دستگاه کنترلی در یک کاربرد خاص (مثلاً صنعتی) به کار گرفته می‌شوند. در این نرم‌افزارها دیر پاسخ دادن به همان بدی پاسخ ندادن است. در این نوع نرم‌افزارها هدف اصلی طراحان، پاسخگویی سریع (در مهلت تعیین شده) به رویدادها و درخواست‌ها می‌باشد و راحتی کاربران و بهره‌وری منابع در درجه‌های بعدی اهمیت، قرار دارند. نتیجه اینکه زمان پاسخ در سیستم‌های بی‌درنگ الزاماً باید به موقع و تضمین شده باشد. در طرف مقابل، در سیستم‌های اشتراک زمانی و عمومی، داشتن زمان پاسخ کوتاه مطلوب است ولی الزامی نیست.

به طور کلی سیستم‌های بی‌درنگ به دو نوع زیر طبقه‌بندی می‌شوند:

۱- سیستم بی‌درنگ سخت (Hard Real-Time)

۲- سیستم بی‌درنگ نرم (Soft Real-Time)

در سیستم‌های بی‌درنگ سخت، ضرب‌العجل‌ها یا مهلت زمانی (deadline) باید تحت هر شرایطی رعایت شود، مانند ترمز اتومبیل، باید گرفته شود و زمان بسیار نزدیک است این بی‌درنگ سخت است، دیر پاسخ دادن به همان بدی پاسخ ندادن است، نباید دیر پاسخ دهد یا دستگاه کنترل ضربان قلب انسان، اگر ضربان نبض نبود همه رو باید سریع بیدار کند و نباید دیر پاسخ دهد. در سیستم‌های بی‌درنگ سخت، معمولاً وسایل ذخیره‌سازی ثانویه همچون دیسک به دلیل کندی آن وجود ندارد و به جای آن از حافظه‌های ROM استفاده می‌شود. سیستم عامل‌های پیشرفته نیز در این سیستم‌ها وجود ندارد چرا که سیستم عامل کاربر را از سخت‌افزار جدا می‌کند و این جداسازی باعث عدم قطعیت در زمان پاسخگویی می‌شود. به دلیل نیاز به پاسخ‌دهی سریع و تضمین شده سیستم‌های بی‌درنگ از حافظه مجازی استفاده نمی‌کنند. در سیستم‌های بی‌درنگ سخت مهلت زمانی (deadline) باید پشتیبانی شود. در برخی کاربردها (مثل کنترل صنعتی) در کامپیوترها از سیستم عامل استفاده نمی‌شود. از آنجا که در سیستم‌های کنترل صنعتی برنامه می‌بایست در اسرع

وقت در مقابل یک اتفاق، از خود عکس العمل نشان دهد، وجود واسط سیستم عامل باعث کند شدن مراحل می گردد. البته در سیستم های بی درنگ سخت می تواند سیستم عامل باشد، اما سیستم عامل باید با شرایط سیستم های بی درنگ سخت سازگار باشد یعنی سیستم عامل هم بی درنگ باشد. سیستم عامل بی درنگ نوعی سیستم عامل است که در آن، زمان پارامتر کلیدی است. سیستم بی درنگ به سیستمی گفته می شود که درستی اجرای یک عملیات در آن تنها به درست بودن عملیات از نظر منطقی بستگی نداشته باشد بلکه اجرای آن عملیات در یک بازه زمانی مشخص نیز در درستی اجرای عملیات در نظر گرفته شود. در سیستم های بی درنگ سخت (hard real-time) یا به عبارتی سیستم های بی درنگ بدون وقفه (immediate real-time) پایان اجرای یک عملیات پس از ضرب الاجل بی فایده تلقی می شود و نوبتدارو پس از مرگ سهراب است.

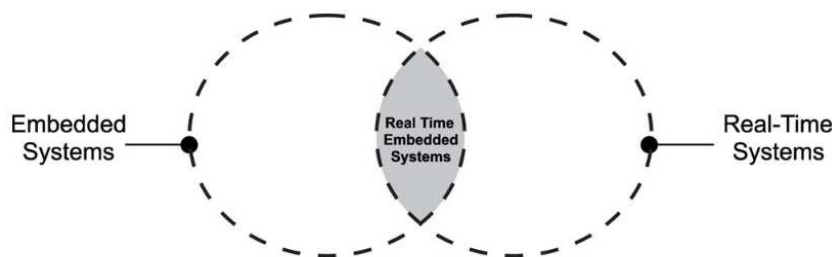
اما در سیستم های بی درنگ نرم، زیر پا گذاشتن ضرب الاجل ها با اینکه نامطلوب است اما قابل تحمل است. یعنی می توان با چند لحظه تاخیر نیز کنار آمد. مثل رزرو بلیط هواپیما در صورت خالی شدن لحظه ای یک صندلی، خوب است اولویت و حق رعایت شود و فوراً رزرو انجام شود، ولی اگر هم ضرب الاجل ها گاهی رعایت نشد خیلی هم فاجعه بار نیست و چنین تاخیری قابل تحمل است یا هشدار پیام خالی شدن کاغذ یک دستگاه فتوکپی، خوب است پیام خالی شدن کاغذ فوراً اعلام شود، ولی اگر هم ضرب الاجل ها گاهی رعایت نشد خیلی هم فاجعه بار نیست و چنین تاخیری قابل تحمل است. سیستم های پخش زنده صدا و تصویر نیز معمولاً سیستم های بی درنگ نرم هستند که در صورت عدم پاسخگویی سیستم در ضرب الاجل با پایین آوردن کیفیت صدا و تصویر وضعیت را مدیریت می کنند. نتیجه اینکه در سیستم های بی درنگ نرم، رعایت مهلت زمانی مطلوب است، ولی اجباری نیست و تضمین هم نمی شود. به بیان دیگر سیستم تلاش می کند که کار خود را در مهلت زمانی خاص انجام دهد، ولی اجباری هم در این کار نیست. و حتی انجام کار پس از پایان مهلت زمانی، بازهم معنا دارد. پس در سیستم بی درنگ سخت احتمال تاخیر زمانی تا پس از مهلت زمانی وجود دارد. اما در سیستم بی درنگ نرم احتمال تاخیر زمانی تا پس از مهلت زمانی وجود دارد.

توجه: می توان گفت یک سیستم بی درنگ سخت تضمین می کند که کارها و وظایف بحرانی به موقع انجام شود، اما در یک سیستم بی درنگ نرم، یک وظیفه بحرانی نسبت به سایر وظایف اولویت خیلی بالاتری دارد و تا پایان تکمیل شدنش این ارجحیت را حفظ می کند. در واقع در سیستم های بی درنگ سخت پس از پایان مهلت زمانی، ادامه و تکمیل یک کار، دیگر معنی ندارد. از آنجا که سیستم های بی درنگ نرم مهلت زمانی (deadline) را پشتیبانی نمی کنند، استفاده آنها در کنترل صنعتی ریسک آور است. هر چند که سیستم های بی درنگ نرم می بایست پاسخی سریع داشته باشند ولی مساله پاسخدهی به حادی سیستم های بی درنگ سخت نمی باشد.

انسان در وادی زندگی نیازهای گوناگونی دارد، یکی از نیازهای اساسی انسان، نیاز به امنیت است. اما گاهی، ممکن است در معرض عوامل محیطی و بیرونی و یا حتی درونی امنیت انسان در

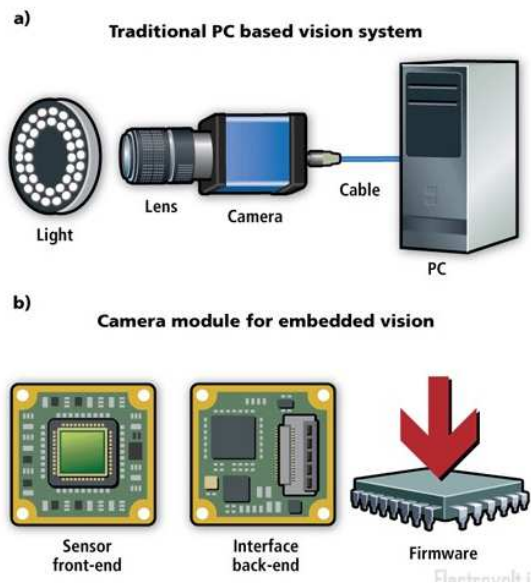
شرایط هشیاری یا ناهشیاری به مخاطره بیفتد. بنابراین نیاز است تا مکانیزمی همواره هوشیار و همیشه بیدار و با اشراف لحظه به لحظه، مخاطرات پیرامون انسان را رصد و تحت کنترل خود قرار دهد تا در موقع لزوم و به صورت آنی، بی‌درنگ، در لحظه و در زمان حقیقی و واقعی (تا دیر نشده) با تهدید مقابله کند، نرم‌افزارهای بی‌درنگ این نگهبان همیشه هوشیار و همیشه بیدار هستند. مانند نرم‌افزارهای ترمز اتومبیل، کنترل ضربان قلب اتاق بیهوشی، کنترل فشار کابین هواپیما و ...

در عصر حاضر دو مفهوم جدید وجود دارد که باعث بوجود آمدن نسل جدیدی از سیستم‌های پرکاربرد شده است. این دو مفهوم یکی «سیستم‌های نهفته» یا Embedded Systems و دیگری «سیستم‌های بی‌درنگ» یا Real-Time System می‌باشند. در اغلب اوقات این دو سیستم به صورت تلفیقی و تحت عنوان «سیستم‌های نهفته بی‌درنگ» یا Real-Time Embedded Systems مورد استفاده قرار می‌گیرند. این سیستم‌ها توانایی کنترل دامن‌های وسیعی از وسایل مکانیکی و الکترونیکی را دارا می‌باشند. اگر به اطراف خود نگاهی بیاندازید، انواع مختلف آنها را مشاهده می‌کنید. تلویزیون، لوازم منزل و آشپزخانه، تلفن‌های همراه هوشمند، سیستم‌های کنترل اتومبیل، سیستم‌های کنترل ترافیک، سیستم‌های اتوماسیون صنعتی، ربات‌ها، موشک‌های نظامی و ... همه و همه مثال‌هایی از این سیستم‌ها می‌باشند. سیستم‌های نهفته که به آنها سیستم‌های تعبیه شده یا توکار نیز گفته می‌شود، سیستم‌های کامپیوتری هستند که شامل اجزای الکترونیکی و یا مکانیکی می‌باشند و وظیفه‌ی مشاهده (Monitor)، پاسخ دادن (Respond) و کنترل (Control) محیط خارجی سیستم را بر عهده دارد. این محیط خارجی بوسیله دستگاه‌های ورودی و خروجی نظیر سنسورها (Sensors)، عمل‌کننده‌ها (Actuators) و ... با سیستم‌های نهفته ارتباط دارد. علت نام‌گذاری سیستم‌های نهفته (Embedded Systems) این است که در گذشته با نگاه کردن به درون یک سیستم مثل PC و Laptop نگاه ناظر می‌توانست اجزای فیزیکی داخلی آن نظیر CPU و RAM را ببیند، اما در سیستم‌های نهفته این اجزا درون IC هستند و روی برد الکترونیکی به نوعی پنهان شده است و به صورت مجزا قابل مشاهده نیست.



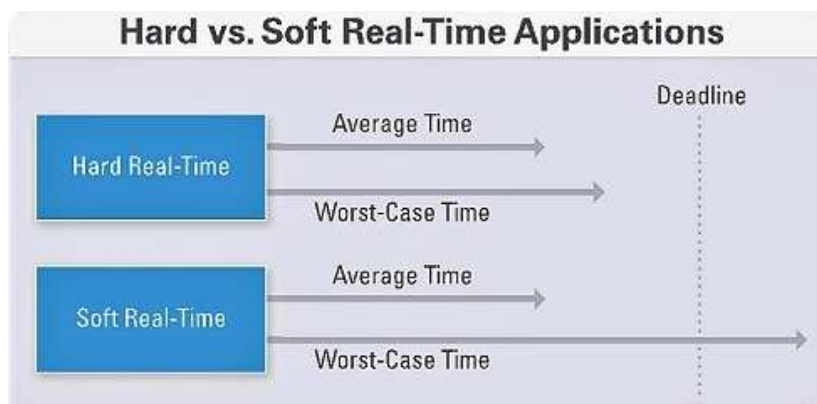
بر خلاف کامپیوترهای همه منظوره (به عنوان مثال کامپیوترهای شخصی) که برای رفع نیازهای عمومی طراحی شده‌اند، سیستم‌های نهفته به گونه‌ای طراحی می‌شوند که برای یک کاربرد خاص با کمترین هزینه، بهترین کارایی را از خود نشان دهند. بنابراین مشخصه‌ی کلیدی، سیستم‌های نهفته، طراحی اختصاصی برای انجام یک کار مشخص است. از آنجاکه سیستم‌های

نهفته برای یک کار مشخص اختصاص یافته‌اند، مهندسین طراح می‌توانند محصول را برای کاهش اندازه و قیمت بهینه کرده و اطمینان‌پذیری و کارایی آنرا بالا ببرند، برخی از سیستم‌های نهفته با بهره‌گیری از مزیت‌های تولید با تیراژ بالا و به تبع مقرون به صرفه بودن هزینه‌های تولید، به شکل انبوه تولید شده‌اند. امروزه درون اکثر وسایل و دستگاه‌های پیرامون ما (خودپرداز، تلفن همراه، اتومبیل و ماشین لباسشویی) سیستم نهفته قرار دارد. شکل زیر یک سیستم امنیتی را در دو شکل نهفته و PC Based (مبتنی بر PC) نشان می‌دهد.



در علم کامپیوتر، محاسبات بی‌درنگ (RTC: Real-Time Computing) و یا محاسبات واکنشی (Reactive Computing)، سیستم‌های نرم‌افزاری و سخت‌افزاری را توصیف می‌کند که زمان در آنها اهمیت دارد و بایستی این تضمین را حاصل کنند که خروجی صحیح سیستم، در یک بازه‌ی مشخصی حتماً تولید شود، چرا که تولید این خروجی در خارج از این بازه زمانی، حتی اگر صحیح نیز باشد، دیگر مطلوب نخواهد بود. این آستانه‌های زمانی، سرحد زمانی یا Deadline نامیده می‌شود. پاسخ‌های سیستم‌های بی‌درنگ به رویدادها اغلب در حدود میلی ثانیه و میکروثانیه هستند.

همانطور که گفتیم سیستم‌های بی‌درنگ در عمل بر دو نوع تقسیم‌بندی می‌شوند. یکی سیستم‌های بی‌درنگ نرم (Soft) و دیگری سیستم‌های بی‌درنگ سخت (Hard) که در شکل زیر قابل مشاهده است:



در یک سیستم بی‌درنگ سخت، زمان پاسخ به رویداد هم در حالت میانگین و هم در بدترین حالت هر دو کمتر از زمان Deadline هستند، اما در سیستم‌های بی‌درنگ نرم، زمان پاسخ به رویداد فقط در حالت میانگین کمتر از زمان Deadline است، و در بدترین حالت حتی ممکن است بیشتر از زمان Deadline باشد و از Deadline هم عبور کند. در حقیقت هنگام طراحی یک سیستم بی‌درنگ از نوع سخت باید دقیق و سخت‌گیرانه عمل کرد، اما در طراحی سیستم بی‌درنگ از نوع نرم نیازی به این سخت‌گیری زیاد نمی‌باشد.

مفهوم سیستم بی‌درنگ نهفته (Real Time Embedded Systems)، یک مفهوم علمی است که علوم متفاوتی را الزاماً در بر می‌گیرد. این سیستم‌ها، همان‌طور که از نامش پیداست، تلفیقی از دو سیستم نهفته و بی‌درنگ است. یک سیستم نهفته بی‌درنگ را معمولاً اینگونه تعریف می‌کنند، «سیستمی که محیط را از طریق دریافت داده‌ها، پردازش آنها و سپس برگرداندن سریع تاثیر پردازش این داده‌ها به محیط، کنترل می‌کند».

توجه: در سیستم عامل‌های عمومی و همه منظوره یا GPOS (General Purpose Operating System) کاربر هیچ گونه کنترلی بر تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها ندارد، ولی در سیستم عامل‌های بی‌درنگ یا RTOS (Real Time Operating System) به خصوص بی‌درنگ سخت لازم است به کاربر به طور وسیع و گسترده اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها داده شود.

صورت سوال به این شکل است:

در سیستم‌های تعبیه شده بی‌درنگ سخت که پاسخ در زمانی مشخص باید تضمین شود، کدام روش نگاشت ریسمان‌های کاربر به ریسمان‌های سیستمی، مناسب است؟

(۱) یک به یک

گزینه اول پاسخ سوال نیست، زیرا در روش یک به یک فقط زمان‌بند پردازنده و زمان‌بند چندنخی در سطح هسته وجود دارد و زمان‌بند چند نخی در سطح کاربر در این روش مورد

استفاده قرار نمی‌گیرد. در واقع هسته سیستم عامل نه تنها فرآیندها را می‌شناسد بلکه از وجود نخ-های داخل یک فرآیند نیز در صورت وجود نخ آگاه است. در واقع مدیریت نخ‌ها و زمان‌بند چندنخی (نخ‌های فرآیندهای کاربر و نخ‌های فرآیندهای سیستم عامل) در سطح هسته، توسط هسته سیستم عامل انجام می‌گردد و کاربر هیچ دیدی از این کار ندارد. در سیستم‌های بی‌درنگ به خصوص بی‌درنگ سخت لازم است به کاربر به طور وسیع و گسترده اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها داده شود. در روش یک به یک، سطح هسته وجود دارد و به تبع همروندی (در سیستم‌های تک‌پردازنده‌ای) پس از مسدود شدن یک نخ و توازی (در سیستم‌های چندپردازنده‌ای) وجود دارد که این امر منجر به این می‌شود که زمان پاسخ کوتاه باشد و پاسخ قبل از اتمام مهلت زمانی تولید شود. اما در روش یک به یک، سطح کاربر وجود ندارد، که این امر منجر به این می‌شود که اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها به طور وسیع و گسترده به کاربر داده نشود.

(۲) چند به یک

گزینه دوم پاسخ سوال نیست، زیرا در روش چند به یک فقط زمان‌بند پردازنده و زمان‌بند چند نخی در سطح کاربر وجود دارد و زمان‌بند چند نخی در سطح هسته در این روش مورد استفاده قرار نمی‌گیرد. در واقع هسته سیستم عامل فقط فرآیندها را می‌شناسد و هیچ اطلاعاتی از نخ‌ها ندارد. در واقع اولویت‌بندی نخ‌ها، مدیریت نخ‌ها و زمان‌بند چندنخی در سطح کاربر و توسط یک بسته نرم‌افزاری انجام می‌گردد. بدین معنی که نخ‌ها را برنامه‌نویس مشخص می‌کند و مدیریت آن-ها را نیز بر عهده می‌گیرد. در سیستم‌های بی‌درنگ به خصوص بی‌درنگ سخت لازم است به کاربر به طور وسیع و گسترده اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها داده شود. در روش چند به یک، سطح هسته وجود ندارد و به تبع همروندی (در سیستم‌های تک-پردازنده‌ای) پس از مسدود شدن یک نخ و توازی (در سیستم‌های چندپردازنده‌ای) وجود ندارد که این امر منجر به این می‌شود که زمان پاسخ طولانی شود و به تبع این احتمال وجود دارد که پاسخ بعد از اتمام مهلت زمانی تولید شود. هرچند که در روش چند به یک، سطح کاربر وجود دارد، که این امر منجر به این می‌شود که اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها به طور وسیع و گسترده به کاربر داده شود.

(۳) چند به چند

گزینه سوم پاسخ سوال نیست، زیرا روش چند به چند از اجتماع دو روش سطح کاربر و هسته ابداع گردیده است. در روش چند به چند علاوه بر زمان‌بند پردازنده و زمان‌بند چندنخی در سطح هسته سیستم عامل، زمان‌بند چند نخی نیز در سطح کاربر برای زمان‌بندی و اولویت‌دهی نخ‌های فرآیند کاربر وجود دارد. در واقع زمان‌بندی نخ‌های فرآیندهای سیستم عامل، توسط هسته سیستم عامل و زمان‌بندی نخ‌های فرآیندهای کاربر، توسط برنامه کاربر انجام می‌گردد، که این امر منجر به اولویت‌بندی نخ‌های فرآیندهای کاربر می‌گردد. در این روش، زمان‌بند چند نخی در سطح کاربر،

نخ‌های کاندید خود را از میان نخ‌های متعدد در فرآیندهای مختلف کاربر براساس یک الگوریتم خاص انتخاب و تحویل یک زمان‌بند چند نخی در سطح هسته می‌دهد. در واقع انتخاب نخ‌های کاندید (زمان‌بندی) در فرآیندهای کاربر به خود کاربر واگذار شده است که همانطور که گفتیم منجر به اولویت‌بندی نخ‌های فرآیندهای کاربر می‌گردد.

در سیستم‌های بی‌درنگ به خصوص بی‌درنگ سخت لازم است به کاربر به طور وسیع و گسترده اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها داده شود. در روش چند به چند، سطح هسته وجود دارد و به تبع همروندی (در سیستم‌های تک‌پردازنده‌ای) پس از مسدود شدن یک نخ و توازی (در سیستم‌های چندپردازنده‌ای) وجود دارد که این امر منجر به این می‌شود که زمان پاسخ کوتاه باشد و پاسخ قبل از اتمام مهلت زمانی تولید شود. اما یک ریسک دارد اینکه در روش چند به چند برنامه‌نویس به تعداد دلخواه نخ ایجاد می‌کند، اما ممکن است تعداد LWP کمتری به آنها منتسب شود تا سربار هسته کمتر شود. زیرا در روش چند به چند نخ‌های سطح کاربر به تعداد کم‌تر یا مساوی از نخ‌های سطح هسته نگاشت می‌شود. برای مثال اگر یک فرآیند پنج درخواست (نخ) همزمان داشته باشد، اگر چهار LWP داشته باشیم، یکی از درخواست‌ها (نخ‌ها) تا بازگشت نتیجه یکی از چهار درخواست (نخ) دیگر از سطح هسته به جریان نخواهد افتاد و این یعنی تاخیر و ممکن است زمان پاسخ طولانی شود و به تبع این احتمال وجود دارد که پاسخ بعد از اتمام مهلت زمانی تولید شود. هرچند که در روش چند به چند، سطح کاربر وجود دارد، که این امر منجر به این می‌شود که اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها به طور وسیع و گسترده به کاربر داده شود.

۴) دوسطحی

گزینه چهارم پاسخ سوال است، زیرا روش دو سطحی به نوعی روش چند به چند پلاس است، یعنی روش دوسطحی بهبود یافته روش چند به چند است، در واقع روش دوسطحی علاوه بر اینکه روش چند به چند را در خود دارد، جهت بهبود کارایی روش یک به یک را نیز در خود قرار داده است. در سیستم‌های بی‌درنگ به خصوص بی‌درنگ سخت لازم است به کاربر به طور وسیع و گسترده اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها داده شود. در روش دوسطحی (Two-Level Model) یا همان چند به چند پلاس، سطح هسته وجود دارد و به تبع همروندی (در سیستم‌های تک‌پردازنده‌ای) پس از مسدود شدن یک نخ و توازی (در سیستم‌های چندپردازنده‌ای) وجود دارد که این امر منجر به این می‌شود که زمان پاسخ کوتاه باشد و پاسخ قبل از اتمام مهلت زمانی تولید شود. اما یک ریسک دارد اینکه در روش چند به چند برنامه‌نویس به تعداد دلخواه نخ ایجاد می‌کند، اما ممکن است تعداد LWP کمتری به آنها منتسب شود تا سربار هسته کمتر شود. زیرا در روش چند به چند نخ‌های سطح کاربر به تعداد کم‌تر یا مساوی از نخ‌های سطح هسته نگاشت می‌شود. برای مثال اگر یک فرآیند پنج درخواست (نخ) همزمان داشته باشد، اگر چهار LWP داشته باشیم، یکی از درخواست‌ها (نخ‌ها) تا بازگشت نتیجه

یکی از چهار درخواست (نخ) دیگر از سطح هسته به جریان نخواهد افتاد و این یعنی تاخیر و ممکن است زمان پاسخ طولانی شود و به تبع این احتمال وجود دارد که پاسخ بعد از اتمام مهلت زمانی تولید شود. اما از آنجا که روش دوسطحی علاوه بر داشتن روش چند به چند، روش یک به یک را نیز دارد، می‌تواند در مواقع لزوم برای پاسخ سریع از روش یک به یک نیز استفاده نماید، زیرا در روش یک به یک هر رشته نخ کاربر به یک رشته نخ هسته نگاشت می‌شود. همچنین در روش دوسطحی (Two-Level Model) یا همان چند به چند پلاس، سطح کاربر وجود دارد، که این امر منجر به این می‌شود که اجازه تعیین اولویت‌ها، مهلت‌ها و کنترل زمان‌بندی فرآیندها و نخ‌ها به طور وسیع و گسترده به کاربر داده شود. روش دوسطحی در سیستم عامل‌هایی نظیر HP-UX، IRIX و True64nIx پشتیبانی می‌شود. نسخه‌های قبل از Solaris9 نیز مدل دوسطحی را پشتیبانی می‌کنند، اما Solaris9 از روش یک به یک استفاده می‌کند.