



فصل دوم: فرایند (Process)

مریم تقی زاده

ترم اول ۱۴۰۴

Email: taghizadehmail@gmail.com

رئوس مطالب فصل ۲

• فرآیند

- تعاریف و مفاهیم اولیه
- حالات فرآیند
- انواع فرآیندها

• نخ

- ارتباط بین فرآیندها
- بررسی مشکلات کلاسیک
- زمانبندی
- بن بست

فرآیند (Process)

پشته

Heap

داده

کد برنامه

- برنامه در حالت معمول نهاد غیرفعال است.
- فرآیند:
 - برنامه در حال اجرا را گویند.
 - چون فرآیند در حال اجرا است نیاز به منابعی دارد مانند حافظه، CPU، دستگاه ورودی/خروجی
 - علاوه بر کد برنامه، شامل
 - مقدار شمارنده برنامه، ثبات های CPU، پشته و بخش داده ها (Data Section)
 - کد برنامه را بخش متن می نامند. (Text Section)

تعاریف

- فرآیند CPU-Limited

- بیشتر زمان کامپیوتر صرف محاسبات می شود و CPU بیشتر درگیر است

- فرآیند I/O-Limited

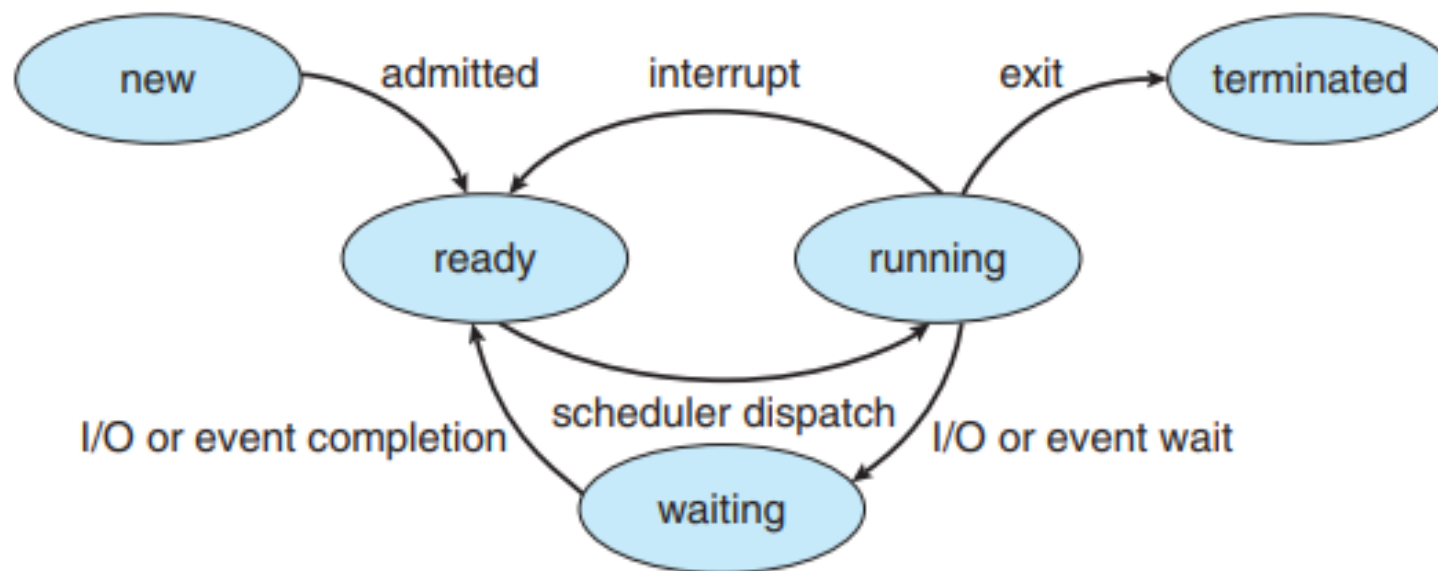
- بیشتر زمان کامپیوتر صرف دستگاههای I/O (ورود و خروج داده ها) می شود.

تعاریف

- فرآیند وقتی که در حالت اجرا قرار دارد می تواند در حالت های مختلفی قرار گیرد.
- حالت فرآیند : توسط فعالیت فعلی آن تعریف می شود.
- جدید (New): فرآیند، جدیداً ایجاد شده است.
- اجرا (Running): دستورات فرآیند در حال اجرا است.
- انتظار (Waiting) یا مسدود (Blocked): فرآیند منتظر یک رویدادی است که رخ دهد.
- آماده (Ready): فرآیند آماده است تا CPU به آن اختصاص داده شود تا بتواند اجرا شود.
- خاتمه (Terminated): اجرای فرآیند به اتمام رسیده است.
- در هر لحظه فقط یک فرآیند می تواند در پردازنده اجرا شود.



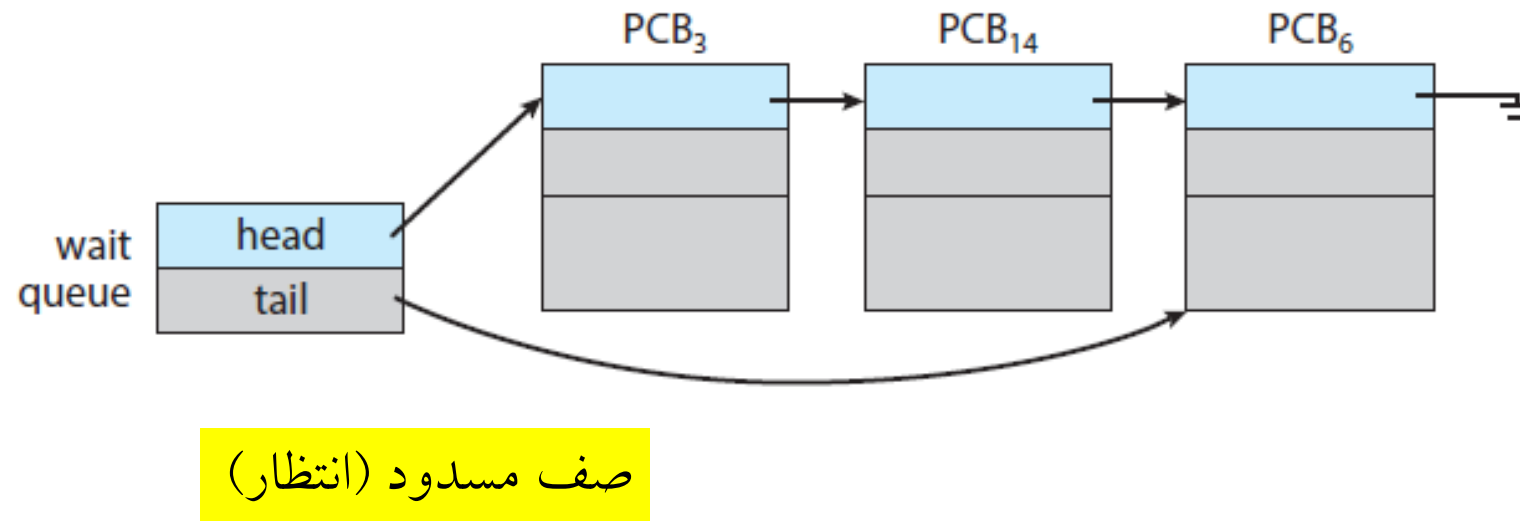
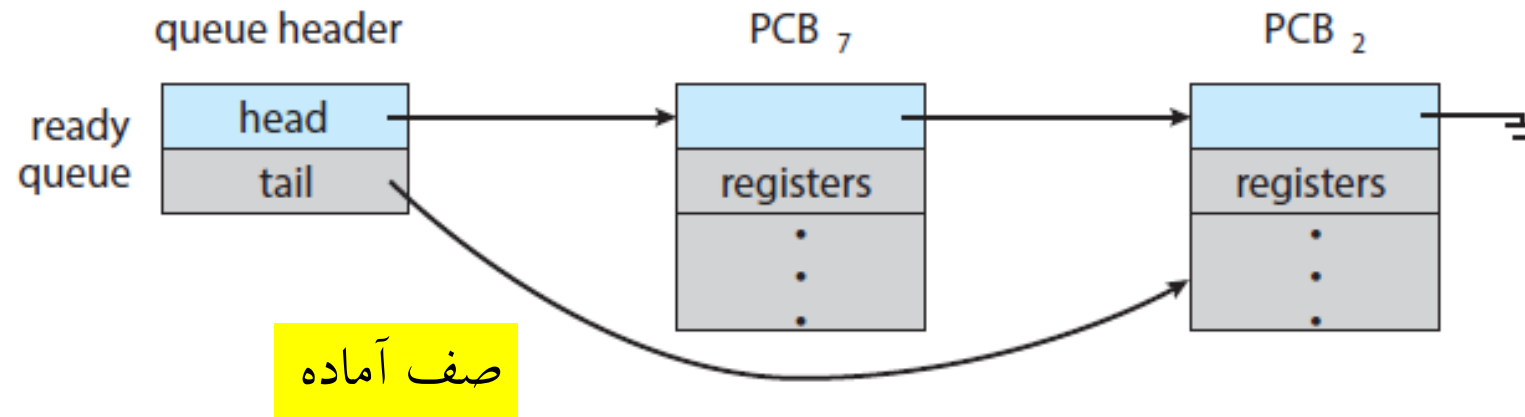
مدل حالات فرآیند (۵ حالت)



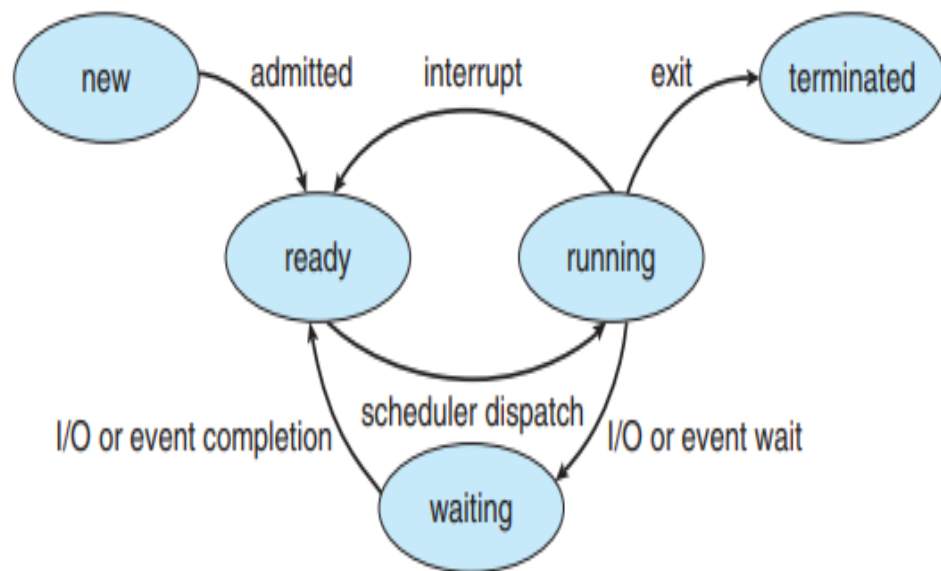
تمام حالات ارائه شده در این تصویر، در حافظه **RAM** قرار دارند.



صف آماده و انتظار



تغییر حالات فرآیند در مدل ۵ حالت



- جدید به آماده
- آماده به اجرا
- اجرا به خروج
- مسدود (انتظار) به آماده
- اجرا به مسدود (انتظار)
- اجرا به آماده
- آماده به خروج
- مسدود (انتظار) به خروج

تغییر حالات فرآیند در مدل ۵ حالت

زمانی که سیستم عامل یکی از فرآیندهای آماده را به حالت اجرا می برد، به این عمل توزیع (dispatch) می گویند.

زمانی که فرآیند از حالت انتظار (مسدود)، با رخ دادن رویداد به حالت آماده می رود، به آن **wake up** گویند.

به دلیل اتمام بازه زمانی اجرای فرآیند در سیستم چندبرنامه ای، یا روی دادن وقفه، فرآیند از اجرا به آماده می رود.

آماده به خروج (مسدود به خروج): با خاتمه فرآیند والد، ممکن است همه فرآیندهای فرزند نیز پایان یابد.

مدل حالات فرآیند (۷ حالت)

• معلق (Suspend)

- اگر تمام فرآیندهای در مرحله اجرا منتظر I/O باشند (که در حالت انتظار یا مسدود قرار خواهند گرفت)، این امکان وجود دارد که:
- CPU بیکار بماند.
- صف آماده تقریباً خالی باشد.

• تکنیک مبادله (Swapping)

- فرآیند از حافظه اصلی به حافظه دیسک (یا بالعکس) منتقل می شود. (Swap out/In)
- خارج یا داخل کردن فرآیندها از حافظه، باعث می شود که فرآیندهای وابسته به I/O که فعلاً CPU نیاز ندارند از حافظه خارج و بالعکس، فرآیندهای وابسته به CPU که به CPU نیاز دارند در حافظه وارد شوند.

این حالت به منظور جلوگیری از هدر رفتن زمان پردازنده و استفاده حداکثری از حافظه فرض شده است.



مدل حالات فرآیند (۷ حالت)

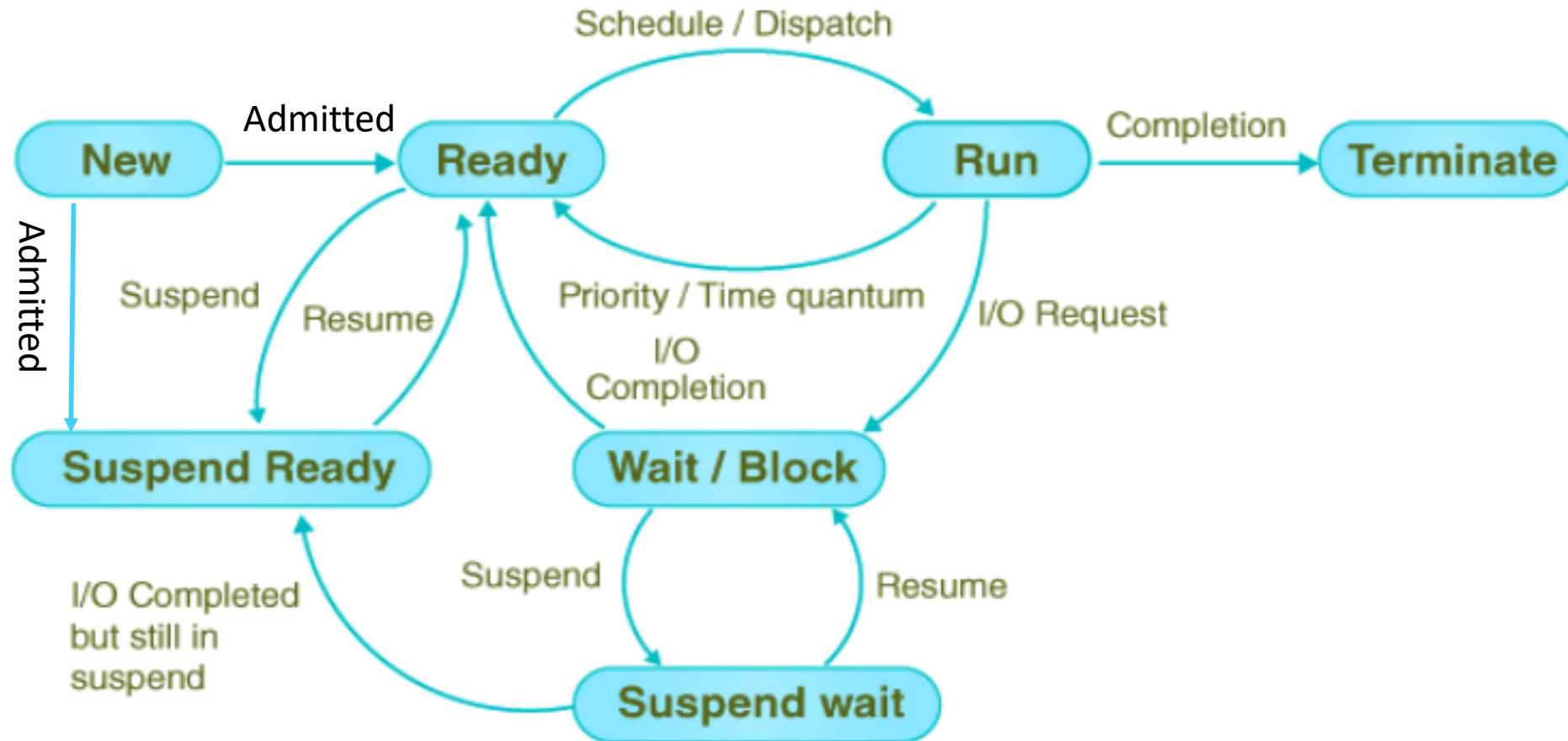
• حالت مسدود و معلق (Suspend-Block یا Suspend-Wait)

- زمانی که تعداد فرآیندهای مسدود (منتظر) زیاد باشد ممکن است حافظه کم بیاد، سیستم عامل فرآیندها را از انتظار معلق می کند که به عنوان حال «مسدود و معلق» شناخته می شود.
- فرآیند در حافظه ثانویه است و منتظر بروز یک رویداد است.

• حالت آماده و معلق (Suspend-Ready)

- اگر فرآیند آماده باشد ولی حافظه نداشته باشیم به حالت «آماده و معلق» می رود.
- فرآیند در حافظه ثانویه است و به محض بارگذاری در حافظه اصلی، آماده اجرا است.

مدل حالات فرآیند (۷ حالت)



عملیات های بر روی فرآیند

• ایجاد فرآیند

- هنگامی که یک برنامه اجرا می شود یک فرآیند جدید ایجاد می شود.
- سیستم عامل منابع مورد نیاز را به آن فرآیند تخصیص می دهد.
- یک PCB برای ذخیره اطلاعات فرآیند جدید استفاده می شود.
- یک فرآیند (فرآیند والد) می تواند فرآیند جدیدی (فرآیند فرزند) ایجاد کند.

• پایان فرآیند

- با پایان اجرای برنامه، فرآیند خاتمه می یابد.
- سیستم عامل منابع تخصیص یافته به فرآیند را آزاد می کند.

دلایل پایان فرآیند

- پایان نرمال برنامه
- سقف زمانی
- انتظار زیادتر از حد برای بروز یک رویداد
- خطای حافظه (نیاز به حافظه بیش از حد)
- خطای دسترسی به مکان های غیرمجاز حافظه
- خطای ورودی/خروجی
- استفاده نامناسب از داده
- خطای حفاظت (تلاش برای دسترسی به منبعی که مجاز نیست)
- خطای محاسباتی (مانند تقسیم به صفر)
- پایان یافتن فرآیند والد
- درخواست والد
- دستورالعمل نامعتبر
- اجرای دستورالعمل ممتاز (هسته)
- دخالت سیستم عامل

بلوک کنترل فرآیند

- بلوک کنترل فرآیند (Process Control Block)

- یک ساختار داده در سیستم عامل است که اطلاعات مربوط به هر فرآیند در حال اجرا را نگهداری می کند.

- این اطلاعات برای مدیریت و کنترل فرآیندها توسط سیستم عامل ضروری است.

- حاوی اطلاعات زیر است:

- شناسه فرآیند (شماره منحصر بفرد فرآیند)، حالت فرآیند

- شمارنده برنامه، ثبات های CPU

- اطلاعات زمانبندی CPU (اولویت فرآیند، اشاره گر به صف زمانبندی)

- اطلاعات مدیریت حافظه (اطلاعات حافظه تخصیص یافته)

- اطلاعات حسابرسی (میزان استفاده از پردازنده)

- اطلاعات وضعیت I/O

process state
process number
program counter
registers
memory limits
list of open files
• • •

ضرورت بلوک کنترل فرآیند

مدیریت فرآیندها

- به سیستم عامل امکان می دهد تا فرآیندهای مختلف را ردیابی و مدیریت کند.

تعویض متن

- هنگام تعویض بین فرآیندها، سیستم عامل اطلاعات PCB را ذخیره و بازیابی می کند تا اجرای فرآیندها بدرستی ادامه یابد.

زمانبندی فرآیندها

- از اطلاعات PCB در تصمیم گیری در زمانبند سیستم عامل استفاده می شود.

مدیریت منابع PCB

- اطلاعات مربوط به منابع تخصیص یافته به هر فرآیند را نگهداری می کند.

تعویض متن (Context-switching)

- تعویض متن:

- زمانی که سیستم عامل CPU را از یک فرآیند (P_i) گرفته و در اختیار یک فرآیند دیگر (P_j) قرار می دهد، اطلاعات PCB که برای اجرای مجدد فرآیند P_i لازم دارد را نگهداری می کند و اطلاعات PCB فرآیند P_j را بازیابی می کند.

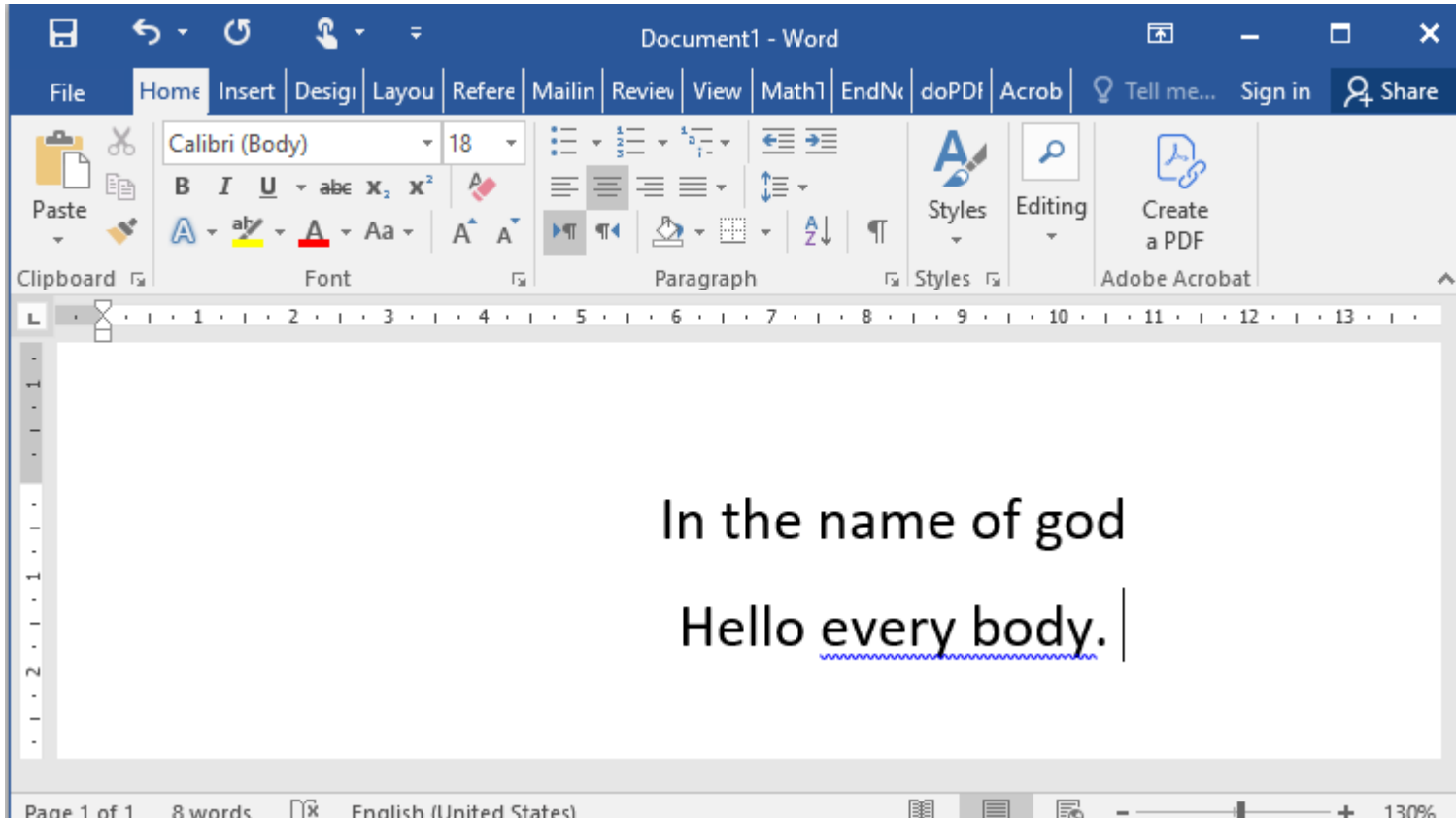
- این تعویض متن توسط هسته سیستم عامل انجام می شود.

- زمان تعویض متن، سربار اضافی برای کامپیوتر است.

نخ (Thread)

- مدل اولیه فرآیند:
 - برنامه در حال اجرا با یک رشته اجرا
 - فرآیند، دستورات را به صورت متوالی اجرا می کند.
 - محدودیت: (۱): عدم توانایی در اجرای همزمان وظایف (۲): عدم بهره‌وری از پردازنده‌های چند هسته‌ای
- رشته، ترد (نخ)
 - واحدهای کوچکتری از اجرا در یک فرآیند هستند.
 - فرآیند می تواند چندین نخ داشته باشد که بطور همزمان اجرا شوند. (اجرای همروند)
 - امتیازات: (۱): اجرای همزمان وظایف (۲): استفاده کارآمدتر از منابع سخت افزاری
- هر فرآیند می‌تواند چند رشته اجرایی داشته باشد. به رشته‌های اجرایی که می‌توانند موازی با هم اجرا شوند، نخ یا ترد گویند.

مثال نخ (Thread)



مثال: پردازشگر داده

- نخ ۱: نوشتن کارکترها
- نخ ۲: بررسی املای کلمات
- نخ ۳: بررسی گرامر
- نخ ۴: ذخیره خودکار فایل

مثال نخ (Thread)

- مثال: صفحات مرورگر وب

- نخ ۱: بارگذاری و نمایش متن وب
- نخ ۲: برای بارگذاری هر تصویر در صفحه وب
Thread21, thread22, thread23, ...
- نخ ۳: بارگذاری ویدئوهای صفحه وب

مثال نخ (Thread)

- اجرای برنامه چند رسانه ای (بخوایم یک فیلم را نمایش بدهیم)

- نخ ۱: مسؤل خواندن فایل

- نخ ۲: پخش صدا و تصویر

- نخ ۳: مدیریت رابط کاربری برنامه



با استفاده از چندنخی، می توان همزمان فایل را خواند و صدا و تصویر را پخش کرد و رابط کاربری را مدیریت کرد.

تفاوت نخ و فرآیند

- فرآیند یک واحد مستقل از اجرای برنامه است که دارای فضای آدرس، منابع سیستم و شناسه منحصر بفرد است.
- نخ یک واحد اجرایی درون فرآیند است که فضای آدرس، منابع سیستم و شناسه فرآیند را با سایر نخ های موجود در همان فرآیند به اشتراک می گذارد.

ویژگی نخ

- نخ‌ها دارای شناسه، شمارنده برنامه، ثبات‌ها و پشته مخصوص به خود هستند.
- نخ‌ها می‌توانند **همزمان اجرا** شوند و به این ترتیب، عملکرد برنامه را بهبود بخشند.
- نخ‌ها می‌توانند با یکدیگر **ارتباط** برقرار کنند و داده‌ها را به **اشتراک** بگذارند.

امتیازات نخ

- پاسخگو بودن

- اگر قسمتی از برنامه بلوکه شود و یا عملیات طولانی داشته باشد، بقیه نخ های برنامه اجرا می شوند.

- اشتراک منابع

- نخ ها در حافظه و منابعی که مربوط به فرآیند آن ها است، شریک هستند.

- صرفه اقتصادی

- تخصیص حافظه و منابع برای فرآیند هزینه بر است. اما به دلیل اشتراک منابع یک فرآیند برای نخ های درون آن، ایجاد نخ مقرون به صرفه است.

- توسعه پذیری

- استفاده از نخ در سیستم های چندپردازنده ای می تواند قابلیت اجرای موازی را فراهم نماید.

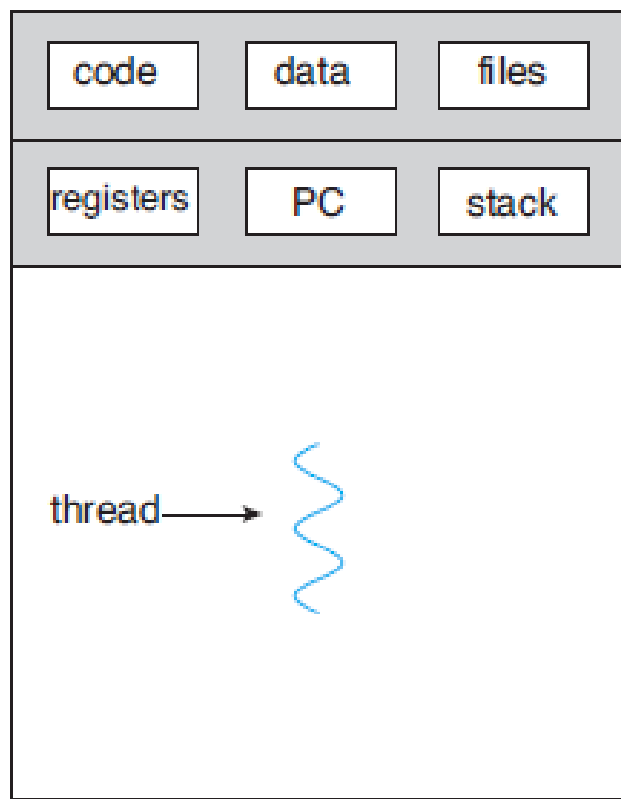
فرآیند تک نخه و چندنخه

- فرآیندهای تک نخه و چندنخه دو مدل اصلی در سیستم عامل برای اجرای برنامه ها هستند.

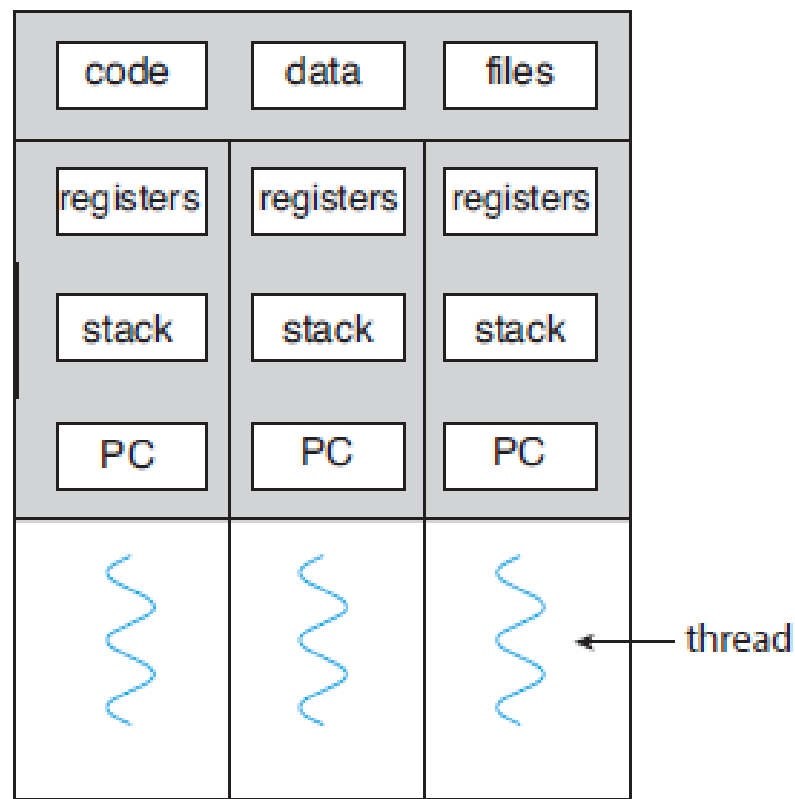
- تک نخه (Single thread)

- چند نخه (Multi thread)

فرآیند تک نخه و چندنخه



فرآیند تک نخه



فرآیند چند نخه



در یک فرآیند با چندین نخ، نخ‌ها در قسمت کد، داده، فایل‌ها و منابع سیستم عامل شریک هستند.

فرآیند تک نخ و چندنخی

• تک نخ (Single thread)

- هر فرآیند فقط یک نخ اجرایی داد.
- دستورات به صورت متوالی و پشت سر هم اجرا می شوند.
- اگر دستور یا وظیفه ای مسدود شود، کل فرآیند مسدود می شود.
- پیاده سازی ساده ای دارد.
- برای برنامه ساده تک وظیفه ای مناسب است.

• معایب

- عملکرد پایین تر به دلیل عدم امکان اجرای همزمان وظایف.
- مسدود شدن یک دستور، کل نخ و بنابراین فرآیند را مسدود می کند.

فرآیند تک نخ و چندنخی

- چند نخ (Multi thread)
 - هر فرآیند می تواند چندین نخ اجرایی داشته باشد.
 - وظایف می توانند همزمان یا موازی اجرا شوند.
 - پیاده سازی و اشکال زدایی آن پیچیده است.
 - برای برنامه های چندوظیفه ای مناسب می باشد.
 - استفاده بهینه از منابع
 - اجرای همزمان عملکرد بالایی را به همراه دارد
- معایب
 - مدیریت نخ های متعدد سبب مصرف منابع بیشتر می شود.
 - به دلیل دسترسی همزمان به منابع مشترک مشکلات همزمانی را به دنبال دارد.

انواع نخ

- نخ در سطح کاربر (User thread)
 - نخ های کاربر در سطح کاربر مدیریت می شوند.
 - سیستم عامل عملی برای نخ های کاربر انجام نمی دهد و اطلاعی از وضعیت نخ ها ندارد.
 - نخ کاربر توسط کتابخانه های کاربر مدیریت می شود (شامل ایجاد و حذف و مدیریت و زمانبندی نخ).
- نخ در سطح هسته (Kernel thread)
 - نخ های هسته سیستم عامل به وسیله سیستم عامل پشتیبانی و اجرا می شوند.
 - این نخ ها در سطح سیستم ایجاد می شوند.

مقایسه نخ کاربر و هسته

• نخ کاربر

• مزایا:

- سرعت بالا: تعویض بین نخ های کاربر نیاز به تعامل با هسته ندارد.
- هزینه پایین: ایجاد و مدیریت نخ های کاربر کم هزینه است.
- قابل حمل بودن: نخ کاربر می تواند بر روی هر سیستم عاملی که از کتابخانه نخ کاربر پشتیبانی می کند اجرا شود.
- ارتباط و همگام سازی راحتتر است.

• معایب

- مسدود شدن: اگر نخ فراخوانی سیستم نیاز داشته باشد، بدلیل اینکه سیستم عامل اطلاعی از نخ ها ندارد، پس کل فرآیند با تمام نخ های آن مسدود می شود، اگر نقص حافظه رخ دهد، کل فرآیند مسدود می شود.
- مدیریت توسط کاربر سبب می شود که هسته بدلیل نا آگاهی از نخ ها، بخوبی تخصیص منابع بین نخ ها را انجام ندهد.
- عدم پشتیبانی از چند پردازنده ای: توزیع نخ ها و زمانبندی آنها میان چند پردازنده به خوبی انجام نمی شود.

مقایسه نخ کاربر و هسته

• نخ هسته

• مزایا:

- مدیریت توسط هسته: تخصیص بهتر منابع سیستم بین نخ ها توسط هسته انجام شده و زمانبندی اجرا را کنترل می کند.
- مسدود شدن یک نخ سبب، مسدودی دیگر نخ های یک فرآیند نمی شود.
- معایب نخ های سطح کاربر را ندارد.
- پشتیبانی از چند پردازنده و همزمانی واقعی: بهبود عملکرد برنامه های چند نخ و افزایش سرعت را به همراه دارد.

• معایب

- هر عملیات نخ توسط هسته انجام شده و به یک فراخوانی سیستمی نیاز دارد و هزینه بالا است.
- سربار سیستم: تعویض بین نخ های هسته نیاز به تعامل با هسته دارد که سربار سیستم افزایش می یابد.

مقایسه نخ کاربر و هسته



اگر برنامه به عملکرد بالا و پشتیبانی از چند پردازنده نیاز دارد، پیاده سازی نخ هسته مناسب تر است.

اگر برنامه به سرعت بالاتر و انعطاف پذیری نیاز دارد، پیاده سازی نخ کاربر مناسب تر است.

ارتباط نخ کاربر و هسته

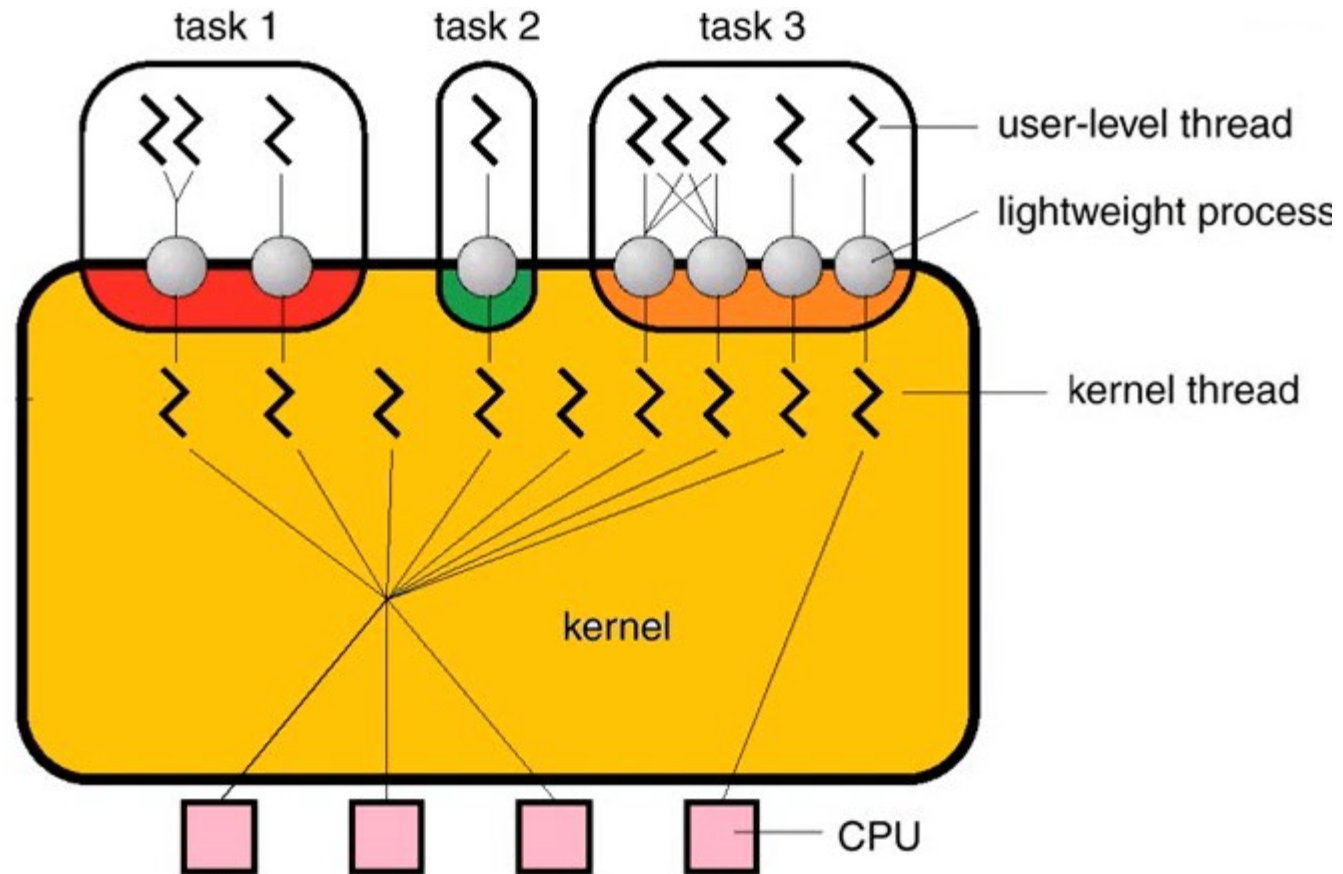
- سیستم عامل خبری از نخ‌های کاربر ندارد. برای این که نخ‌های کاربر بتوانند از امکانات سیستم عامل نیز استفاده کنند، نخ‌ی در هسته سیستم عامل درست می‌شود که با نخ کاربر در ارتباط باشد.

- استفاده از مزایای نخ کاربر و نخ هسته

روش ترکیبی (Hybrid thread)

- روش ترکیبی مزایای هر دو رویکرد نخ های کاربر و نخ های هسته را ترکیب می کند و معایب آن ها را کاهش می دهد.
- عملکرد: نگاشت بین نخ سطح کاربر و سطح هسته
 - تعدادی نخ کاربر به تعدادی نخ هسته نگاشت می شوند.
 - در واقع سیستم عامل تعدادی نخ هسته را برای اجرای نخ های کاربر اختصاص می دهد.
 - فرآیند سبک وزن (Light Weight Process)، نگاشتی بین نخ سطح کاربر و نخ سطح هسته است.
 - هر LWP، یک یا تعداد بیشتری نخ را حمایت می کند و به یک نخ سطح هسته نگاشت می کند.
 - مزایای سیستم ترکیبی،
 - چندین نخ درون یک برنامه می توانند به طور موازی بر روی چند پردازنده اجرا شوند.
 - مسدود شدن یک نخ، مسدودی کامل فرآیند را به دنبال ندارد.
- این روش می تواند ۳ مدل مختلف را ایجاد کند: یک به یک، چند به یک، چند به چند.

روش ترکیبی



روش ترکیبی

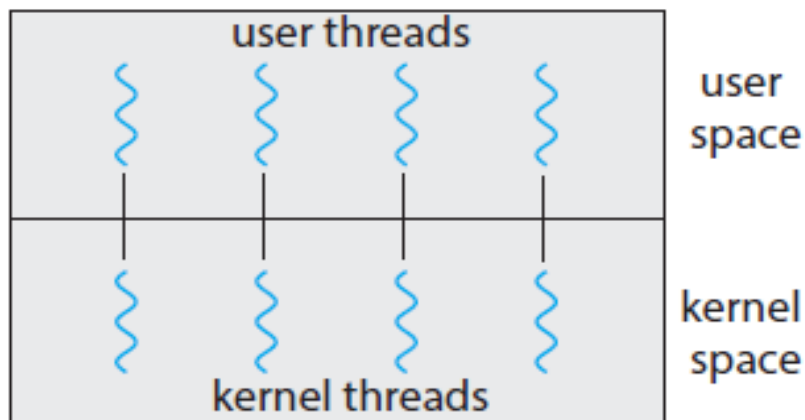
- مدل نخ های ترکیبی تلاش می کند تعادلی بین عملکرد و انعطاف پذیری ایجاد کند. با استفاده از نخ های هسته، امکان بهره مندی از مزایای چند پردازشی واقعی فراهم می شود و با استفاده از نخ های کاربر، تعویض نخ ها سریع تر انجام می شود و پاسخگویی برنامه بهبود می یابد.
- پیاده سازی و مدیریت نخ های ترکیبی پیچیده تر از مدل های نخ های کاربر یا هسته خالص است.



مدل های چند نخي

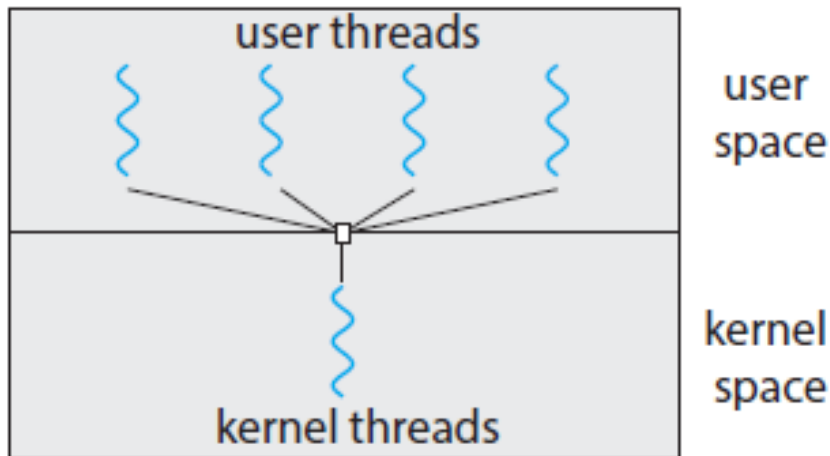
- مدل یک به یک (One-to-One model)

- هر نخ سطح کاربر به یک نخ سطح هسته متصل می شود
- بین نخ های کاربر و نخ های هسته یک رابطه یک به یک وجود دارد.
- امکان اجرای همزمان نخ ها بر روی پردازنده های مختلف وجود دارد.
- مسدود شدن یک نخ سبب مسدود شدن کل فرآیند نمی شود.
- ایجاد نخ های زیاد، سربار زیادی به سیستم تحمیل می کند.

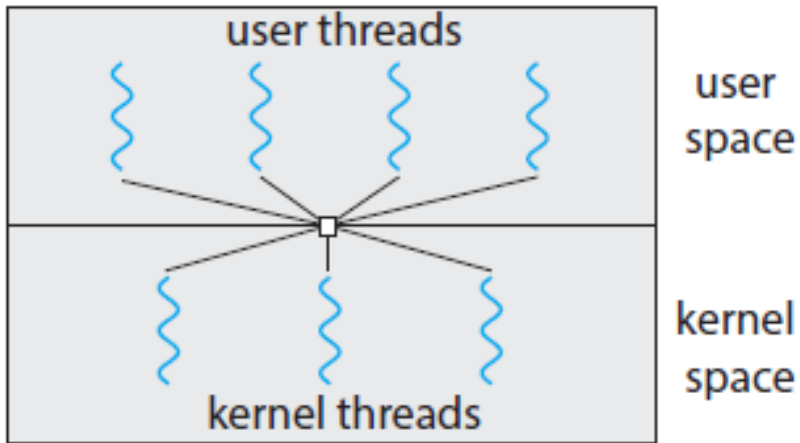


مدل های چند نخي

- مدل چند به یک (Many-to-One model)
 - تعداد زیادی نخ کاربر با یک نخ هسته در ارتباط هستند.
 - مدیریت نخ ها در سطح کاربر انجام می شود که **سریع تر و کارآمدتر** است.
 - قابلیت جابجایی نخ ها بین سیستم عامل های مختلف وجود دارد.
 - مسدود شدن یک نخ کاربر، باعث **مسدود شدن** نخ های مرتبط با آن نخ هسته می شود.
 - امکان استفاده از **چند پردازنده ای وجود ندارد**.



مدل های چندنخی



• مدل چند به چند (Many-to-Many model)

- چندین نخ سطح کاربر به تعداد کمتری نخ سطح هسته نگاشت می شود.
- تعداد نخ های هسته می تواند کمتر یا مساوی تعداد نخ های سطح کاربر باشد.
- در مدیریت نخ ها انعطاف پذیری بالایی وجود دارد.
- موازی سازی نخ ها بر **پردازنده های مختلف امکان پذیر است**.
- یک فراخوان مسدود کننده، کل سیستم را **مسدود نمی کند**.
- امکان استفاده از نخ های زیاد در سطح کاربر، بدون **اعمال بار زیادی بر هسته** فراهم است.
- پیچیدگی پیاده سازی و مدیریت نخ ها، می تواند بالا باشد.
- این احتمال وجود دارد که عملکرد سیستم در شرایطی غیرقابل پیش بینی باشد.