

سیستم عامل جلسه 3

حالت های فرآیند ها

وقتی فرایندی اجرا می شود، حالت آن تغییر می کند. حالت فرایند تا حدی توسط فعالیت فعلی آن تعریف می شود

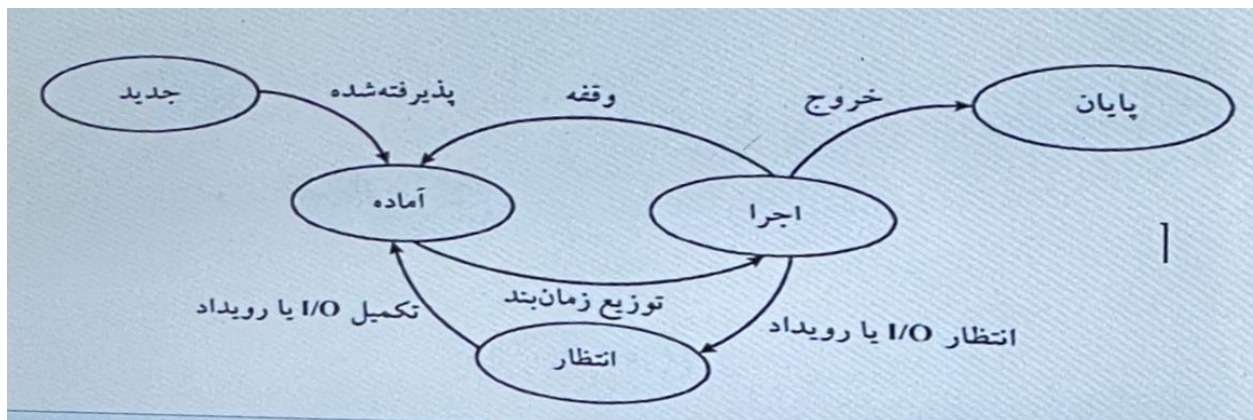
هر فرایند ممکن است در یکی از حالت های زیر باشد

جدید: فرآیند ایجاد می شود

اجرا: دستورات در حال اجرا هستند

انتظار: فرآیند منتظر وقوع رویدادی است (مثل کامل شدن عمل i/o یا دریافت یک سیگنال)

پایان: اجرای فرایند خاتمه یافته است



مفاهیم اساسی زمانبندی

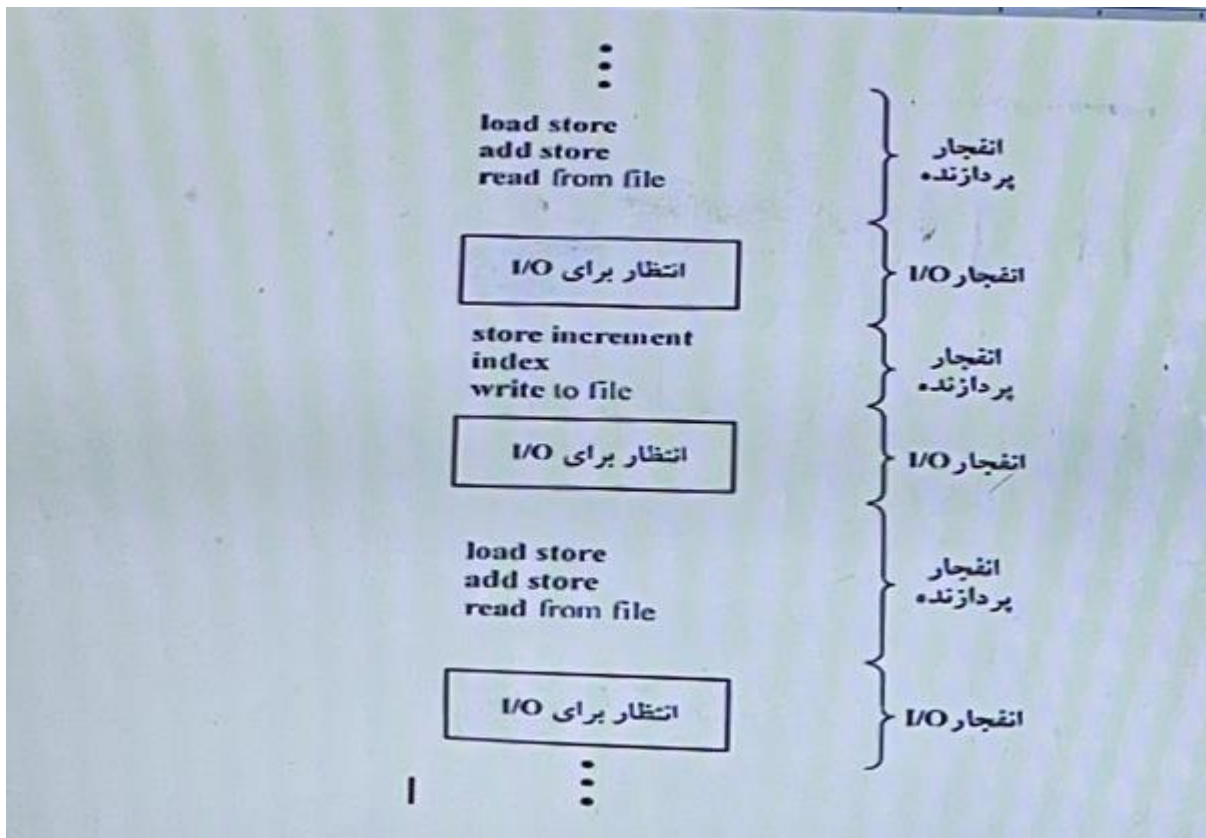
در یک سیستم تک پردازنده ای ، در هر زمان فقط یک فرایند می تواند اجرا شود؛ بقیه باید منتظر بمانند تا پردازنده آزاد شود تا دوباره بتوانند زمانبندی شوند. هدف چند برنامه ای این است که همیشه چندین فرایند در حال اجرا باشند تا بهره وری پردازنده به حداکثر برس. هر فرایند به اجرایش ادامه می دهد تا برای یک عمل I/O در انتظار بماند در یک سیستم کامپیوتری ساده ، در این مدت پردازنده باید بیکار بماند در چندین برنامه ای، سعی می شود که از این زمان به نحوه احسن استفاده شود، در این سیستم ، چند فرایند به طور همزمان در حافظه نگهداری می شوند، وقتی یک فرایند به حالت انتظار می رود. سیستم عامل پردازنده را از آن فرایند می گیرد و به فرایند دیگری می دهد این روند ادامه می یابد ، هر وقت که فرایندی به حالت انتظار رفت، فرایند دیگری از پردازنده استفاده می کند

تقریباً تمام منابع کامپیوتری، قبل از استفاده از زمانبندی می شوند. پردازنده یکی از منابع اصلی کامپیوتر است و در نتیجه زمانبندی آن موضوع اصلی طراحی سیستم عامل است

چرخه انفجار I/O و انفجار پردازنده

موفقیت زمانبندی پردازنده به این خاصیت فرایند ها بستگی دارد که اجرای فرایند شامل چرخه ای از اجرای پردازنده و انتظار برای I/O است . فرایندها بین این دو حالت سوییچ می کنند. اجرای فرایند با انفجار پردازنده شروع می شود. و به دنبال آن یک انفجار I/O قرار دارد که بعد از آن ، انفجار دیگری از پردازنده و سپس انفجار دیگری

از i/o قرار دارد و به همین ترتیب ادامه می یابد . سرانجام، آخرین انفجار پردازنده با درخواست سیستم برای خاتمه اجرا، پایان می پذیرد



مدت این انفجار های پردازنده از فرایندی به فرایند دیگری و از کامپیوتری به کامپیوتر دیگر متفاوت است. تعداد زیادی از انفجار های کوتاه پردازنده و تعداد اندکی از انفجار های بلند پردازنده وجود دارد. برنامه مقید به i/o (در تنگنای i/o) معمولا دارای چند انفجار کوتاه پردازنده است. برنامه مقید به پردازنده (در تنگنای پردازنده) معمولا چند انفجار بلند پردازنده دارد. این توزیع می تواند در انتخال الگوزیتم زمانبندی پردازنده مهم باشد

انواع زمانبندی ها

کلید چند برنامه‌گی زمانبندی است، زمانبندی بر روی کارایی سیستم اثر می‌گذارد زیرا مشخص می‌کند کدام فرایندها منتظر مانده و کدام فرایندها به جلو بروند

انواع زمانبندی برای پردازنده عبارت است از:

1. زمانبند بلند مدت (Long Term scheduler)

تصمیم‌گیری در مورد افزودن به مجموعه‌ی فرایندها برای اجرا

2. زمانبند میان مدت (Middel Term scheduler)

تصمیم‌گیری در مورد افزودن به تعداد فرایندهایی که بخش یا تمام آن‌ها در حافظه اصلی است.

3. زمانبند کوتاه مدت (Short Term Scheduler)

تصمیم‌گیری در مورد این که کدام یک از فرایندهای موجود در حافظه اصلی برای اجرا توسط پردازنده انتخاب شوند

4. زمانبند ورودی – خروجی (i/o Scheduler)

تصمیم‌می‌گیرد که کدام درخواست **i/o** فرایندها به وسیله یک دستگاه **i/o** موجود انجام گیرد.

وظیفه فعال سازی و تعلیق فرایندها بر عهده زمانبند میان مدت است

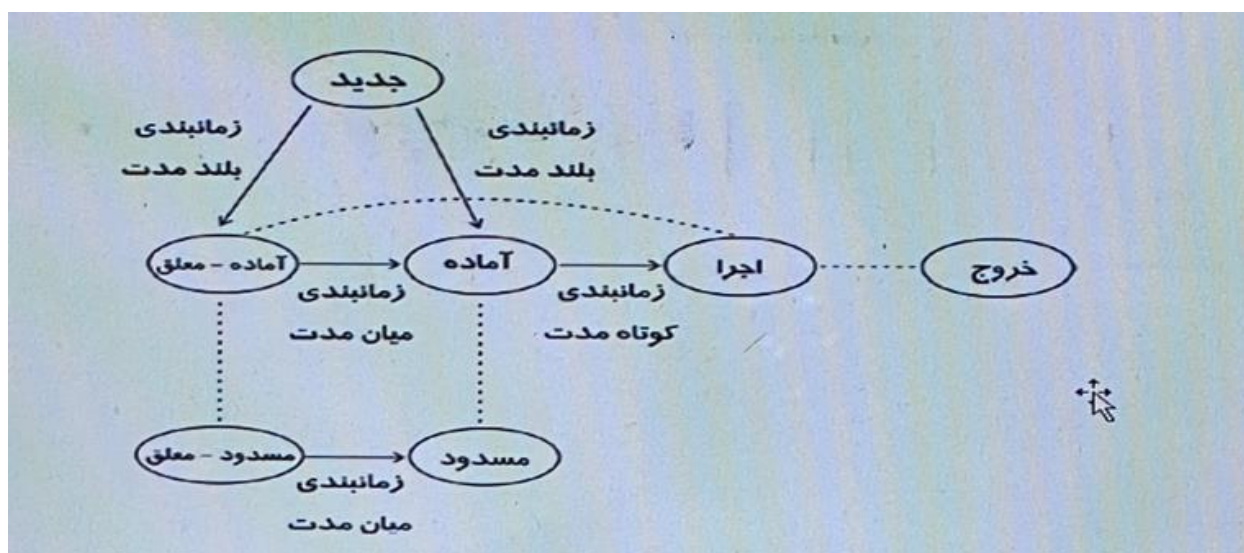
زمانبند میان مدت فرایندی را از حافظه اصلی حذف و به حافظه جانبی می برد. این فرایند بعداً می تواند به حافظه اصلی لود شود. این الگو را مبادله (Swapping) می گویند.

ایده اصلی زمانبندی میان مدت این است که می تواند فرایندی را از حافظه حذف کند و درجه چند برنامگی را کاهش دهد.

زمانبند بلند مدت ترکیب خوبی از فرایندهای i/o limited و CPU limited ، انتخاب می کند. نام دیگر زمانبند بلند مدت ، زمانبند کار است. نام دیگر زمانبند کوتاه مدت، زمانبند پردازنده است.

زمانبند بلند مدت نسبتاً دفعات کمی اجرا می شود. زمانبند میان مدت نسبتاً تعداد دفعات بیشتری به اجرا در می آید و زمانبند کوتاه مدت بیشترین دفعات اجرا را دارد.

تذکر : اکثر سیستم های اشتراک زمانی فاقد زمانبند بلند مدت می باشند.

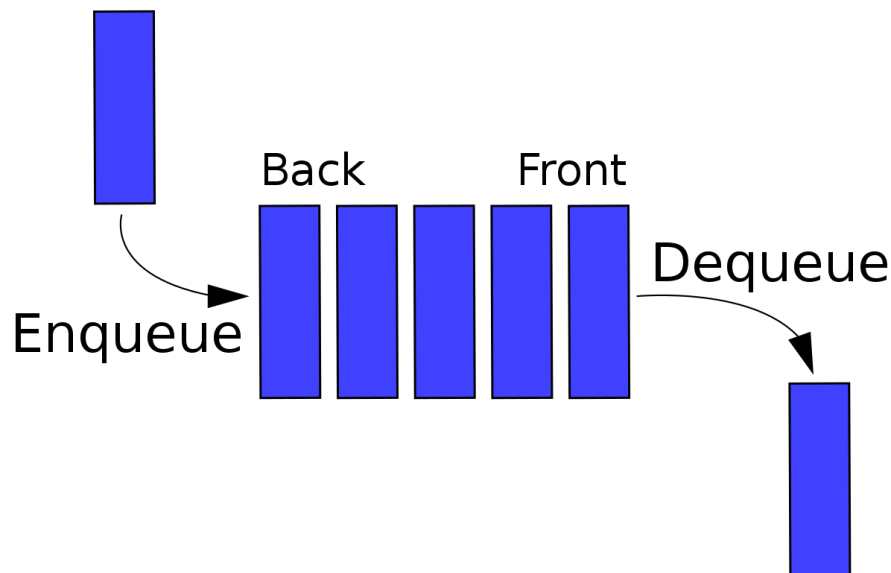


زمانبندی پردازنده

هر وقت پردازنده بیکار می شود، سیستم عامل باید یکی از فرایندهای موجود در صف آماده را برای اجرا انتخاب کند. این انتخاب توسط زمانبند کوتاه مدت (زمان بند پردازنده) انجام می گیرد.

زمان بند، فرایندی را از بین فرایندهای موجود در حافظه که آمادگی اجرا دارند انتخاب می کند و پردازنده را به آن تخصیص می دهد.

صف الزاماً یک صف با ویژگی "خروج به ترتیب" (FIFO) نیست. همان طور که هنگام بحث در مورد الگوریتم های زمانبندی خواهیم دید، صف آماده می تواند به صورت صف (FIFO)، صف اولویت، درخت یا یک لیست پیوندی نامرتب پیاده سازی شود. از نظر مفهومی تمام فرایندهای موجود در صف آماده، منتظر به دست آوردن پردازنده هستند تا اجرا شوند



زمان بندی با قبضه کردن (Preemptive) یا غیر

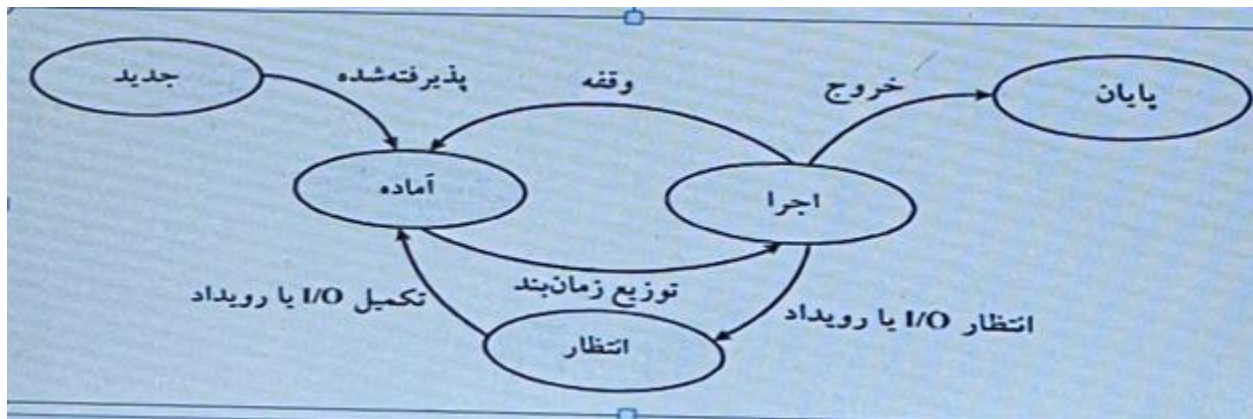
انحصاری یا انقطاع پذیر

فرایند در حال اجرا می تواند به وسیله سیستم عامل متوقف شود و به حالت آماده منتقل شود.

در این الگوریتم ها در پایان برش زمانی و یا تغییر شرایط سیستم مدیر زمانبندی می تواند کنترل پردازنده را از یک "پردازه ی (فرایند)" در حال اجرا گرفته و به "پردازه ی (فرایند)" دیگری بدهد

تصمیمات زمانبندی پردازنده ممکن است تحت چهار شرط ریز اتخاذ شود:

1. وقتی که فرایندی از حالت اجرا به حالت انتظار می رود (مثل در خواست i/o ، یا فراخوانی `wait()` برای خاتمه ی یکی از فرایندهای فرزند)
2. وقتی فرایندی از حالت اجرا به حالت آماده می رود (مثل وقتی که رویدادی رخ می دهد)
3. وقتی که فرایندی از حالت انتظار به حالت آماده می رود (مثل تکمیل شدن i/o)
4. وقتی فرایندی خاتمه می یابد.



برای شرط 1 و 4 انتخابی بر حسب زمانبندی وجود ندارد. در صورت وجود فرایندی در صف آماده، یک فرایند باید برای اجرا انتخاب شود. اما برای شرایط 2 و 3 امکان انتخاب وجود دارد.

وقتی زمانبندی فقط تحت شرایط 1 و 4 انجام می‌گیرد الگوی زمانبندی را **بدون قبضه کردن (NonPreemptive)** می‌نامیم و در غیر این صورت آن را با **قبضه کردن (Preemptive) یا غیر انحصاری** می‌نامیم.

انحصاری / بدون قبضه کردن

در این الگوریتم‌ها تا زمان خاتمه پردازنده و یا زمان نیاز به عمل I/O نمی‌توان CPU را از پردازش در حال اجرا گرفت و پردازش دیگری داد. همین که یک فرایند در حالت اجرا قرار گرفت آن قدر به اجرا ادامه می‌دهد تا خاتمه یابد یا با این که خودش (داوطلبانه) برای انتظار I/O مسدود شود.

