

سیستم عامل (۲) و کارگاه

مفاهیم کلی سیستم عامل

به طور کلی نرم افزارهای کامپیوتر به دو گروه تقسیم می‌شوند:

یکی برنامه‌های سیستمی که عملیات کامپیوتر را مدیریت می‌کنند و دیگری برنامه‌های کاربردی. سیستم عامل (operating system=OS) اصلی‌ترین برنامه سیستمی است که به عنوان رابط بین کاربر و سخت افزار کامپیوتر عمل می‌کند.

سیستم عامل دو وظیفه (یا هدف) اصلی دارد:

سیستم عامل استفاده از کامپیوتر را ساده می‌سازد. این بدان معناست که مثلاً کاربر یا برنامه‌نویس بدون درگیر شدن با مسائل سخت افزاری دیسکها به راحتی فایل‌ها را بر روی دیسک ذخیره و حذف کند. این کار در واقع با به کار بردن دستورات ساده‌ای که فراخوان‌های سیستمی (System Calls) را صدا می‌زنند انجام پذیرد.

در صورت عدم وجود سیستم عامل کاربر و یا برنامه‌نویس می‌بایست آشنایی کاملی با سخت افزارهای مختلف کامپیوتر (مثل مونیتر، فلاپی، کی‌بورد و غیره) داشته باشند و روتین‌هایی برای خواندن و یا نوشتن آنها به زبانهای سطح پائین بنویسد. از این جنبه به سیستم عامل با عنوان ماشین توسعه یافته (Extended machine) یا ماشین مجازی (Virtual machine) یاد می‌شود که واقعیت سخت افزار را از دید برنامه‌نویسان مخفی می‌سازد.

وظیفه دوم سیستم عامل مدیریت منابع (Resource Management) می‌باشد، یعنی سیستم عامل باعث استفاده بهینه و سودمند (اقتصادی) از منابع سیستم می‌گردد. منظور از منابع پردازنده‌ها، حافظه‌ها، دیسکها، ماوس‌ها، چاپگرها، فایلها، پورتها و غیره هستند. یک سیستم کامپیوتری منابع نرم افزاری و سخت افزاری بسیار دارد که ممکن است در حین اجراء برنامه لازم باشند، سیستم عامل همانند مدیر منابع عمل کرده و آنها را بر حسب نیاز به برنامه‌های مشخصی تخصیص می‌دهد.

سیستم عامل معمولاً اولین برنامه‌های است که پس از بوت شدن در حافظه بار می‌شود. پس از بار شدن قسمتی از سیستم عامل بطور دائم در حافظه باقی (Resident) می‌ماند. قسمتهای دیگر با توجه به کاربرد کامپیوتر توسط کاربر از دیسک به حافظه آورده می‌شود.

به قسمت اصلی سیستم عامل که وظایف مهم آن را انجام می‌دهد هسته یا Kernel گفته می‌شود. هسته سیستم عامل برنامه‌ای است که در تمامی اوقات بر روی کامپیوتر در حال اجراست.

سیستم عامل و معماری کامپیوتر اثر زیادی بر روی یکدیگر داشته‌اند. یعنی جهت سهولت کار با سخت افزارهای جدید، سیستم عامل‌ها توسعه یافتند و همچنین در اثنای طراحی سیستم عامل‌ها، مشخص شد که تغییراتی در طراحی سخت افزار می‌تواند سیستم عامل‌ها را ساده تر و کارآمدتر سازد.

هر چند که تطبیق نسل‌های کامپیوتر با نسل‌های سیستم عامل کار درستی نیست ولی این تطبیق که در ادامه انجام می‌دهیم علت ایجاد سیستم عامل‌های جدید را مشخص می‌سازد.

تطابق تکنیک‌های سیستم عامل با نسل های کامپیوتر:

در نسل اول کامپیوترها (۵۵-۱۹۴۵) که از لامپ خلأ برای ساخت آنها استفاده می‌شد، زبان‌های برنامه نویسی (حتی اسمبلی) ابداع نشده بودند و سیستم عامل نیز اصلاً وجود نداشت. روند کار به این صورت بود که برنامه نویسان تنها در یک فاصله زمانی مشخص حق استفاده از کامپیوتر بزرگ و گران قیمت را داشتند.

آنها برنامه‌های خود را توسط تخته مدار سوراخدار (و بعدها توسط کارتهای پانچ) و به زبان ماشین به کامپیوتر می‌دادند. اکثر برنامه‌های محاسبات عددی معمولی مانند جداول سینوس و کسینوس بود.

سیستم های دسته ای Batch system

در نسل دوم، کامپیوترها (۶۵-۱۹۵۵) از ترانزیستور ساخته شدند.

طریقه کار با این کامپیوترهای نسل دوم از طریق یک کنسول (Console) بود که تنها اپراتور مخصوص کامپیوتر با آن کار می‌کرد و کاربران به طور مستقیم با این کامپیوترها محاوره (interaction) نداشتند. کاربر ابتدا برنامه خود را به زبان فرترن یا اسمبلی بر روی کاغذ می‌نوشت سپس توسط دستگاه Card punch، برنامه را روی کارت‌های سوراخدار منتقل ساخت.

بعد این دسته کارت تهیه شده که شامل برنامه، داده‌ها و کارتهای کنترل بود به صورت کار (Job) تحویل اپراتور داده می‌شد. اپراتور بعد از اتمام کار قبلی، دسته کارت جدید را به کامپیوتر می‌داد تا برنامه را اجراء کند در انتها خروجی برنامه (که غالباً چاپی بود) را به کاربر تحویل می‌داد سیستم عامل در این کامپیوترهای اولیه ساده بود و وظیفه اصلی آن انتقال کنترل اتوماتیک از یک کار به کار دیگری بود. سیستم عامل همواره مقیم در حافظه بود و در هر لحظه فقط یک برنامه اجراء می‌شد.

هنگامی که اپراتور مشغول گذاشتن نوارها یا برداشتن کاغذهای چاپ شده بود وقت زیادی از این کامپیوترهای گران قیمت به هدر می رفت .

برای رفع مشکل فوق سیستمهای دسته‌ای (Batch System) ابداع شد . یعنی ابتدا یک سبد پر از دسته کارتها در اتاق ورودی جمع آوری می شد , سپس کلیه آنها به وسیله دستگاه کارتخوان یک کامپیوتر کوچک و نسبتاً ارزان (مثل IBM 1401) خوانده شده و بر روی یک نوار ذخیره می گردید . سپس اپراتور نوار را برداشته بر روی کامپیوتر اصلی و گران قیمت که محاسبات را انجام می داد (مثل IBM7094) نصب می کرد . بعد از آن برنامه‌ای را اجراء می کرد (یعنی سیستم عامل) تا اولین کار را از روی نوار برداشته و اجراء کند, خروجی بر روی نوار دیگری نوشته می شد.

پس از اتمام هر کار سیستم عمل به صورت خود کار کار بعدی را از نوار می خواند. پس از اجراء همه برنامه‌ها , اپراتور نوار خروجی را برداشته و دوباره روی کامپیوتر IBM 1401 منتقل می ساخت تا عملیات چاپ خروجی ها به صورت off line انجام شود. به این روش کار offline spooling نیز گفته می شود. بیشتر برنامه‌های نسل دوم به زبان فرترن و اسمبلی برای محاسبات مهندسی و علمی مثل مشتقات جزئی به کار می رفت.

یکی از معایب روش offline- spooling زیاد بودن زمان برگشت (گردش) (turnaround time) است , یعنی تأخیر زمانی مابین تحویل کار و تکمیل کار. همچنین در این سیستم اولویت بندی به معنای واقعی وجود ندارد.

تنها روش بدست آوردن اولویت این بود که نوار کارهای مهم را ابتدا در ماشین اصلی قرار دهند. حتی در این صورت هم باید چندین ساعت صبر می کردند تا خروجی ها ظاهر شوند. همچنین نیاز به سخت افزار اضافی (مثل کامپیوترهای ۱۴۰۱) از دیگر معایب این روش بود.

سیستم های چند برنامه ای Multi programming

در نسل سوم کامپیوترها (۸۰-۱۹۶۵) از مدارات مجتمع (Integrated Circuit=IC) برای ساخت کامپیوترها استفاده شد. به طور کلی برنامه‌ها را می توان به دو دسته تقسیم کرد : یکی برنامه‌ها با تنگنای محاسباتی CPU bound یا CPU Limiter (مانند محاسبات علمی سنگین که بیشتر زمان کامپیوتر صرف محاسبات Cpu می شود و دیگری برنامه های تنگنای I/O (I/O Limited) مانند برنامه های تجاری که بیشتر زمان کامپیوتر صرف ورود داده‌ها و خروج اطلاعات می شود.

یک اشکال مهم سیستم های دسته‌ای این است که وقتی کار جاری برای تکمیل یک عملیات I/O مثلاً بر روی نوار گردان منتظر می‌شود. در این حال CPU بیکار می‌ماند و مجبور است صبر کند تا عملیات I/O به اتمام برسد. در برنامه های CPU Limited این اتلاف وقت اندک است ولی در برنامه های I/O Limited ممکن است حدود ۸۰ تا ۹۰ درصد وقت CPU به هدر برود.

برای رفع این مشکل از تکنیک multiprogramming استفاده می‌شود. بدنی ترتیب که حافظه به چند قسمت تقسیم شده و در هر قسمت یک برنامه مجزا قرار داده می‌شود. وقتی که یک کار برای تکمیل عملیات I/O منتظر می‌ماند، پردازنده به کار دیگری داده می‌شود. اگر تعداد کارهای موجود در حافظه کافی باشد می‌توان CPU را تقریباً صد در صد مشغول نگه داشت.

البته نگهداری همزمان چند برنامه در حافظه نیاز به مدیریت خاص حافظه دارد تا برنامه‌ها بر همدیگر اثر سوء نداشته باشند. لذا مدیریت حافظه بحث مهمی در سیستم عامل می‌باشد.

سیستم spooling

یکی دیگر از ویژگیهای سیستم عامل نسل سوم Spooling (یا On Line Spooling) است که معمولاً همراه چند برنامه‌گی استفاده می‌شود. این کلمه مخفف عبارت (Simultaneous Peripheral Operation on Line) می‌باشد. در این سیستم به جای آنکه کارتها از دستگاه کارت خوان مستقیماً وارد حافظه گردند و توسط CPU پردازش شوند ابتدا کاراکتر به کاراکتر در بافری در حافظه قرار گرفته و سپس به صورت بلوکی بر روی دیسک نوشته می‌شود.

وقتی که برنامه کاربر اجرا می‌شود و از سیستم عامل تقاضای ورودی می‌کند، اطلاعات ورودی به صورت بلوکی و با سرعت زیاد از دیسک خوانده می‌شوند. به طور مشابه هنگامی که برنامه برای خروجی چاپگر را احضار می‌کند، خط خروجی در یک بافر کپی شده و سپس در دیسک نوشته می‌شود. پس اطلاعات خروجی از دیسک بر اساس ترتیب و اولویت در چاپگر چاپ می‌شوند.

در واقع اسپولینگ عمل I/O یک کار را با عمل محاسباتی کار دیگر روی هم می‌اندازد (overlap). در سیستم اسپولینگ در حالیکه ورودی یک کار از دستگاه ورودی خوانده می‌شود، کار دیگری در حال چاپ شدن است، در همین بین حتی کار دیگری می‌تواند در حال پردازش و اجراء

باشد. در اسپولینگ برنامه عملیات ورودی و خروجی اش را متناسب با سرعت دیسک (که سریع است) انجام می دهد و نه متناسب با سرعت کارتخوان یا چاپگر (که خیلی کند هستند).

بنابراین سیستم مذکور باعث استفاده بهینه از CPU و سایل IO می شود و سرعت عمل را بالا می برد. در این سیستم دیگر نیازی به کامپیوترهای ۱۴۰۱، نوار گردانهای اضافی و حمل نوارها (مانند سیستم های دسته ای) نداریم.

بلوک دیاگرام یک سیستم اسپولینگ می تواند به صورت زیر باشد:

۱- سیستم ورودی کاراکترهایی که توسط کارتخوان وارد می شود را در بلوکهایی جمع آوری کرده و به کمک مدیر دیسک این بلوکها را بر روی دیسک می نویسد. در انتهای هر مدارک ورودی اطلاعاتی راجع به آن مدارک (مانند محل آن بر روی دیسک، اولویت، اسم استفاده کننده) به قسمت زمانبند کار فرستاده می شود.

۲- زمانبند کار (Job scheduler) این زمانبند یک لیست از کارهای موجود در ماشین و اطلاعات لازم در مورد مدارک ورودی مورد نیاز هر یک را نگه می دارد. به این لیست انبار کار یا Jobpool یا Joblist نیز گفته می شود. زمانبند کار به پردازنده کار می گوید که کدام کار بعدی را اجرا کند. برای این منظور اطلاعاتی در مورد محل کار و مدارک ورودی آن بر روی دیسک را به پردازنده کار می دهد. همچنین اگر کارهای متعددی منتظر ورود به حافظه باشند و فضای کافی برای همگی در دسترس نباشد، زمانبند کار تعدادی از آنها را انتخاب کرده و به حافظه می آورد.

۳- پردازنده کار (Job processor) کار داده شده را اجراء می کند. این پردازنده محل کامپایلرها و سایر نرم افزارهای سیستم را بر روی دیسک می داند. هنگام اجراء پردازنده کار خروجی های خود را به صورت بلوکی بر روی دیسک می نویسد و مدارک خروجی را تشکیل می دهد. پردازنده کار اطلاعاتی راجع به محل و اولیت مدارک خروجی به زمانبند خروجی می دهد.

۴- زمانبند خروجی (output scheduler) لیستی از مدارکی که باید چاپ شود را نگه می دارد. وقتی که چاپگر آزاد شد، این زمانبند مدارک بعدی را برای چاپ انتخاب کرده و محل مدارک بر روی دیسک را به سیستم خروجی می گوید.

۵- سیستم خروجی بلاکهای خروجی را از روی دیسک خوانده و کاراکتر به کاراکتر (یا خط به خط) آنها را به چاپگر می فرستد.

۶- مدیر دیسک (Disk Manager) که وظایف خواندن و نوشتن یک بلاک بر دیسک، تخصیص یک بلاک خالی روی دیسک و برگرداندن یک بلاک به مجموعه فضای آزاد دیسک را بر عهده دارد. در خواستهای مربوطه به دیسک در یک صف به نام (Disk Transfer Queue)DTQ ذخیره می گردد.

البته هر سیستم اسپولینگ یک هماهنگ کننده (Coordinator) دارد که مسئول زمانبندی پردازش های سیستم و فراهم کردن عملیاتی که جهت همگام کردن بکار می آیند می باشد. این عملیات توسط دو روال انجام می پذیرد:

Wait: پردازش جاری را متوقف کرده و دوباره وارد زمانبند می گردد.

Free: یک پردازش ویژه را جهت زمانبندی، آماده می کند.

بافر کردن امکان می دهد که عمل I/O یک کار با عمل پردازش همان کار همزمان گردد در حالیکه spooling امکان می دهد عملیات I/O و پردازش چندین کار با هم همزمان گردند.

سیستم اشتراک زمانی Time sharing

این سیستم ها از اوایل سالهای ۱۹۷۰ در نسل سوم کامپیوترها معمول شدند. سیستم اشتراک زمانی در واقع تعمیم سیستم چند برنامه‌گی است.

در سیستم های چند برنامه‌گی کاربر ارتباطی با کامپیوتر نداشت و خطایابی برنامه‌ها مشکل بود چرا که زمان برگشت نسبتاً طولانی اجازه آزمایش کردنهای متعدد را نمی داد. در سیستم اشتراک زمانی کاربر به کمک دو ترمینال (Terminal) که شامل کی برد (برای ورودی) و مونیتر (برای خروجی) است با کامپیوتر به صورت محاوره‌ای (interactive) رابطه برقرار می سازد.

کاربر مستقیماً دستوراتی را وارد کرده و پاسخ سریع آن را روی مونیتر دریافت می کند. در این سیستم ها چندین کاربر به کمک ترمینالهایی که به کامپیوتر وصل است همزمان می توانند از آن استفاده کنند.

در سیستم اشتراک زمانی فقط یک پردازنده وجود دارد که توسط مکانیزمهای زمانبندی بین برنامه‌های مختلف کاربرها با سرعت زیاد (مثلاً در حد میلی ثانیه) سوئیچ می‌شود و بنابراین هر کاربر تصور می‌کند کل کامپیوتر در اختیار اوست. در اینجا تأکید بر روی میزان عملکرد کاربر است یعنی هدف فراهم کردن وسایل مناسب برای تولید ساده نرم افزار و راحتی کاربرد می‌باشد و نه بالا بردن میزان کاربرد منابع ماشین.

کاربر می‌تواند در هر زمان دلخواه برنامه خود را آغاز یا متوقف سازد و یا برنامه را به صورت قدم به قدم اجرا و اشکال زدایی (debug) کند. سیستم‌های دسته‌ای برای اجرای برنامه‌های بزرگ که نیاز محاوره‌ای کمی دارند مناسب است ولی سیستم‌های اشتراک زمانی برای مواردی که زمان پاسخ کوتاه لازم است، استفاده می‌شوند.

در زمانی که کاربری در حال تایپ برنامه‌اش یا فکر کردن روی خطاهای برنامه‌اش می‌باشد CPU به برنامه کاربر دیگری اختصاص یافته تا آن را اجرا کند.

سیستم عاملهای کامپیوترهای شخصی و شبکه

سال ۱۹۸۰ تاکنون که مدارات مجتمع با مقیاس بزرگ (Large Scale Integrated Circuit) ابداع شدند، به عنوان نسل چهارم کامپیوترها شناخته می‌شود. در این سالها کامپیوترهای شخصی با قیمتی ارزان و کارآیی بالا و محیط گرافیکی و محاوره‌ای بسیار خوب به سرعت گسترش یافتند. سیستم عاملهای اولیه بر روی pcها مانند DOS فقط تک کاربره و تک برنامه‌ای بودند.

ولی سیستم عاملهای امروزی آن مانند Windows NT خاصیت‌های چندبرنامگی، چند کاربرته (multiuser) و شبکه‌ای را دارا هستند. با توجه به هزینه اندک سخت افزار اهداف سیستم عامل در طول زمان تغییر کرده است و برای PCها به جای ماکزیمم کردن درصد استفاده CPU و وسایل جانبی، سیستم به سمت راحتی کاربر پیش می‌رود.

به تدریج ویژگی‌های مهم سیستم عاملهای قدیمی در کامپیوترهای بزرگ (مانند حافظت حافظه، حافظه مجازی، محافظت فایلها، همزمانی پردازشها...) بر روی سیستم‌های PC نیز پیاده سازی شده است.

هنگامی که کامپیوترها از طریق شبکه به هم وصل شوند. به آنها ایستگاههای کاری (Work stations) می‌گویند. در یک سیستم عامل شبکه، کاربران از وجود ماشین‌های مختلف در شبکه با

خبرند. آنها می‌توانند از دور وارد یک ماشین شوند و همچنین فایل‌های یک ماشین را روی ماشین دیگر کپی کنند.

هر کامپیوتر سیستم عامل محلی خودش را اجراء می‌کند و کاربر یا کاربران محلی مخصوص به خود را دارد.

سیستم های توزیع شده Distributed system

سیستم عامل توزیع شده در یک محیط شبکه‌ای اجراء می‌شود. در این سیستم قسمتهای مختلف برنامه کاربر بدون آنکه خود او متوجه شود می‌توانند همزمان در چند کامپیوتر مجزا اجراء شده و سپس نتایج نهایی به کامپیوتر اصلی کاربر برگردند.

کاربران نباید از این موضوع باخبر شوند که برنامه آنها در کجا به اجراء در می‌آید و یا فایل‌های آنها در کجای شبکه قرار دارد و همه این کارها باید توسط سیستم عامل به صورت خودکار انجام گیرد. به عبارتی دیگر سیستم باید از دید کاربر شفاف باشد و هرچیز را با نام آن فراخوانی کند و کاری به آدرس آن نداشته باشد.

یکی از مزایای مهم سیستم‌های توزیع شده سرعت بالای اجراء برنامه‌هاست چرا که یک برنامه همزمان می‌تواند از چندین کامپیوتر برای اجراء شدنش استفاده کند.

همچنین به علت توزیع شدن اطلاعات، بانک‌های اطلاعاتی حجیم می‌توانند روی یکسری کامپیوترهای شبکه شده قرار بگیرند. و لازم نیست که همه اطلاعات به یک کامپیوتر مرکزی فرستاده شود (که در نتیجه این نقل و انتقالات حجیم زمان زیادی به هدر می‌رود).

به علت تأخیرهای انتقال در شبکه و نویزهای احتمالی در خطوط انتقالی قابلیت اعتماد اجراء یک برنامه در یک سیستم تنها، بیشتر از قابلیت اجراء آن در یک سیستم توزیع شده است.

همچنین در سیستم توزیع شده اگر یکی از کامپیوترهایی که وظیفه اصلی برنامه جاری را برعهده دارد خراب شود کل عمل سیستم مختل خواهد شد. از طرف دیگر اگر اطلاعاتی همزمان در چند کامپیوتر به صورت یکسان ذخیره گردد و یکی از کامپیوترها خراب شود، داده‌ها را می‌توان از کامپیوترهای دیگر بازیابی کرد از این نظر امنیت افزایش می‌یابد.

به سیستم های توزیع شده گاهی اوقات سیستمهای Loosely Coupled یا ارتباط ضعیف نیز می گویند، چرا که هر پردازنده کلاک و حافظه مستقلی دارد. پردازنده ها از طریق خطوط مخابراتی مختلفی مثل گذرگاه های سریع یا خطوط تلفن ارتباط دارند.

سیستم های چند وظیفه ای Multi tasking

در تکنیک چندنخی (multitasking) یک فرایند (process) که برنامه ای در حال اجراست، می تواند به بخشها یا نخهایی (بندهایی) تقسیم شود که می توانند به صورت همزمان اجرا شوند.

برنامه هایی که چند وظیفه مستقل از هم را انجام می دهند می توانند به صورت چند نخ نوشته شوند. گاهی اوقات به سیستمهای multithreading سیستمهای چند تکلیفی یا چند وظیفه ای (multitasking) هم گفته می شود.

فرآیند (process) یا پردازش اساس یک برنامه در حال اجراست که منابعی از سیستم به آن تخصیص داده شده است (شامل رجیسترها، حافظه، فایلها و دستگاها). فرآیند می تواند مجموعه ای از یک یا چند نخ باشد.

به نخ، رشته یا بند هم گفته می شود. کلیه اطلاعات مربوط به هر پروسس، در یکی از جداول سیستم عامل به نام جداول process Control Block=PCB ذخیره می شود. این جدول یک آرایه یا لیست پیوندی از ساختارهاست که هر عضو آن مربوط به یکی از پروسسهاست که در حال حاضر موجودیت دارد.

اطلاعات موجود در PCB عبارتند از:

حالت جاری پردازش

شماره شناسایی پردازش

اولیت پردازش

نشانی حافظه پردازش

نشانی محل برنامه پردازش بر روی دیسک

نشانی سایر منابع پردازش

محلی برای حفظ ثباتها.

سیستم های چند پردازنده‌های Multi processing

کامپیوترها می‌توانند به جای یک CPU چندین CPU داشته باشند که در اینصورت به آنها سیستم multiprocessing می‌گویند. جهت استفاده از این سیستم‌های نیاز به یک سیستم عامل خاص می‌باشد که بتواند چندین برنامه یا بخش‌های یک فرآیند (را به صورت موازی واقعی روی آنها اجراء کند .

سیستم عامل multitasking برای اجراء چند نخ بر روی یک CPU و سیستم عامل multiprocessing برای اجرای چند نخ بر روی چند CPU به کار می‌روند.

در سیستم چند پردازنده‌ای، CPUها باید بتوانند از حافظه، امکانات ورودی و خروجی و گذرگاه Bus سیستم به صورت اشتراکی استفاده کنند. مزایای این سیستم‌های عبارتند از:

زیاد شدن توان عملیاتی (throughput). منظور از throughput تعداد کارهایی است که در یک واحد زمانی تمام می‌شوند. بدیهی است هر چقدر تعداد پردازنده‌ها بیشتر باشد تعداد کارهای تمام شده در یک پریود زمانی نیز بیشتر خواهد بود. البته این نسبت خطی نیست، مثلا اگر تعداد پردازنده‌ها n باشد سرعت اجراء برنامه‌ها n برابر نمی‌شود چرا که بخشی از وقت پردازنده‌ها جهت مسائل کنترلی و امنیتی و سوئیچ کردنها به هدر می‌رود.

صرفه جویی در هزینه‌ها، از آنجا که پردازنده‌ها منابع تغذیه، دیسکها، حافظه‌ها و ادوات جانبی را به صورت مشترک استفاده می‌کنند در هزینه‌های سخت افزاری صرفه جویی می‌شود.

تحمل پذیری در برابر خطا (fault-tolerant) سیستم‌های مالتی پروسسور قابلیت اعتماد را افزایش می‌دهند چرا که خرابی یک CPU سبب توقف سیستم نمی‌شود بلکه تنها سبب کند شدن آن خواهد شد. استمرار عمل با وجود خرابی نیازمند مکانیزمی است که اجازه دهد خرابی جستجو شده، تشخیص داده شده و در صورت امکان اصلاح شود (یا کنار گذاشته شود). این توانایی به ادامه سرویس، متناسب با سطح بقای سخت افزار، تنزل مطبوع یا graceful degradation نامیده می‌شود.

سیستمهای عملهای چند پردازنده‌ای به دو دسته کلی متقارن و نامتقارن تقسیم می‌شوند.

در سیستم چند پردازنده‌ای نامتقارن (Asymmetric Multi Processing = ASMP) یک پردازنده جهت اجراء سیستم عامل و پردازنده‌های دیگر جهت اجراء برنامه‌های کاربران استفاده می‌شود. از آنجا که کد سیستم عامل تنها روی یک پروسسور اجراء می‌شود، ساخت این نوع سیستم عامل نسبتاً ساده است و از تعمیم سیستم عامل تک پردازنده‌ای به دست می‌آید.

این نوع سیستم عامل‌ها برای اجراء روی سخت افزارهای نامتقارن مناسب هستند، مانند کمک پردازنده و پردازنده‌ای که به هم متصل هستند یا دو پردازنده‌ای که از تمام حافظه موجود مشترکاً استفاده نمی‌کنند. یکی از معایب سیستم عامل نامتقارن غیر قابل حمل بودن (non-portable) آن است. یعنی برای سخت افزارهای مختلف باید سیستم عملهای مختلفی نوشته شود چرا که نامتقارنی می‌تواند حالات مختلف داشته باشد.

در سیستم چند پردازنده‌ای متقارن (symmetric Multi Processing = ASMP) سیستم عامل می‌تواند روی هر یک از پروسسورهای آزاد یا روی تمام پردازنده‌ها همزمان اجراء شود. در این حالت حافظه بین تمام آنها مشترک می‌باشد. تمام پردازنده‌ها اعمال یکسانی را می‌توانند انجام دهند. سیستم متقارن از چند جنبه نسبت به نوع نامتقارن برتری دارد:

از آنجا که سیستم عامل خود یک پردازش سنگین است اگر فقط روی یک CPU اجراء شود باعث می‌گردد که آن پردازنده همواره بار سنگینی داشته باشد، در حالیکه احتمالاً پردازنده‌های دیگر بی‌کار هستند لذا اجراء سیستم عامل روی چند پردازنده باعث متعادل شدن (balancing) بار سیستم می‌شود.

در سیستم نامتقارن اگر پردازنده اجراء کننده سیستم عامل خراب شود کل سیستم خراب می‌شود ولی در سیستم متقارن از این نظر امنیت بیشتر است چرا که اگر یک پردازنده از کار بیفتد سیستم عامل می‌تواند روی پردازنده‌های دیگر اجراء شود.

بر عکس سیستم عامل نامتقارن، سیستم عامل قابل حمل (portable) بر روی سیستم‌های سخت افزاری مختلف است.

سیستم عامل SUNOS ورژن ۴ از نوع نامتقارن و سیستم عامل Solaris2 ورژن و همچنین windows NT از نوع متقارن می‌باشند.

وجود پردازنده‌های متعدد از دید کاربر مخفی است و زمانبندی نخها (Thread) یا فرآیندها (process) روی هر یک از پردازنده‌ها به عهده سیستم عامل است.

گرچه multiprocessing و multithreading امکانات مستقلی هستند ولی معمولاً با هم پیاده سازی می‌شوند. حتی در یک ماشین تک پردازنده‌ای، چند نخ کارایی را افزایش می‌دهد. همچنین ماشین چند پردازنده‌ای حتی برای فرآیندهای غیر نخ هم کارآمد است.

گاهی اوقات به سیستمهای چند پردازنده‌ای، سیستمهای Tightly Coupled یا ارتباط محکم نیز گفته می‌شود. چرا که پردازنده‌ها کلاک (Clock)، گذرگاه و همچنین حافظه مشترکی دارند.

سیستم های بی درنگ Real Time

سیستمهای بی درنگ معمولاً به عنوان یک کنترل کننده در یک کاربرد خاص استفاده می‌شوند. سیستم در این حالت می‌بایست در زمانی مشخص و معین حتماً جواب مورد نظر را بدهد.

سیستمهای کنترل صنعتی، پزشکی، کنترل موشک و غیره از این دسته‌اند.

در سیستمهای بی درنگ زمان پاسخ باید سریع و تضمین شده باشد ولی در سیستم اشتراک زمانی مطلوبست که زمان پاسخ سریع باشند (ولی اجباری نیست). در سیستم دسته‌ای هیچ محدودیت زمانی در نظر گرفته نمی‌شود.

در سیستمهای بی درنگ معمولاً وسایل ذخیره سازی ثانویه وجود ندارد و به جای آن از حافظه های ROM استفاده می‌شود. سیستم عاملهای پیشرفته نیز در این سیستمها وجود ندارند چرا که سیستم عامل کاربر را از سخت افزار جدا می‌کند و این جدا سازی باعث عدم قطعیت در زمان پاسخگویی می‌شود.

سیستمهای بی درنگ با سیستمهای اشتراک زمانی تناقض دارند لذا نمی‌توانند هر دو توأمأ وجود داشته باشند. به دلیل نیاز به پاسخ دهی سریع و تضمین شده سیستمهای بلادرنگ از حافظه مجازی و اشتراک زمانی استفاده نمی‌کنند.

به این سیستمها «بی درنگ سخت» نیز گفته می‌شود.

در سیستمهای «بی درنگ نرم» یک وظیفه بی درنگ بحرانی، نسبت به سایر وظایف اولیت دارد و تا پایان تکمیل شدنش این ارجحیت را دارا خواهد بود. از آنجا که این سیستمها مهلت

زمانی (deadline) را پشتیبانی نمی کنند استفاده آنها در کنترل صنعتی ریسک آور است . هر چند که این سیستمهای بی درنگ نرم می بایست پاسخی سریع داشته باشند ولی مساله پاسخ دهی به حادی سیستمهای بی درنگ سخت نمی باشد.

از کاربردهای سیستم بی درنگ نرم می توان رزرواسیون شرکتهای هواپیمایی , چند رسانه ای (multimedia) واقعیت مجازی (Virtual reality) را نام برد. این سیستمها به ویژگی های سیستم عاملهای پیشرفته (که توسط بیدرنگ سخت حمایت نمی شوند) نیازمندند . بعضی از نسخه های UNIX مانند solaris 2 خاصیت بیدرنگ نرم را دارا می باشند.

در برخی کاربردها (مثل کنترل صنعتی) در کامپیوترها از سیستم عامل استفاده نمی شود. از آنجا که در سیستمهای کنترل صنعتی برنامه می بایست در اسرع وقت در مقابل یک اتفاق , از خود عکس العمل نشان دهد , وجود واسطه سیستم عامل باعث کند شدن مراحل می گردد.

ساختار سیستم عامل:

خدمات و مولفه های سیستم عامل:

مدیریت پردازش در سیستم عامل

مدیریت حافظه در سیستم عامل

CPU بطور مستقیم تنها با حافظه اصلی (Main Memory) سرو کار دارد و برنامه ها جهت اجراء می بایست در حافظه اصلی قرار گیرند.

سیستم عامل در یک سیستم برنامه ای باید مشخص کننده بخش از حافظه توسط چه پروسیسی استفاده شود. تخصیص و باز پس گیری فضاهای حافظه و نیز محافظت از تداخل فرایندها بر یکدیگر از دیگر وظایف سیستم عامل است .
 از آنجا که حافظه اصلی برای جا دادن تمام برنامه های در حال اجراء غالباً کوچک است سیستم عامل بایستی از حافظه ثانویه (Secondary Memory) (عموماً هارد دیسک) جهت پشتیبانی حافظه اصلی استفاده کند.

به این مفهوم حافظه مجازی گفته می‌شود و مدیریت حافظه مجازی، تخصیص و رهاسازی این حافظه و حفاظت آن از دیگر بحثهای سیستم عامل است.

مدیریت فایل در سیستم عامل

جهت استفاده ساده از اطلاعات کامپیوتر، سیستم عامل دید منطقی یکسانی از اطلاعات ذخیره شده روی انواع وسایل ذخیره سازی مثل هارد دیسک، فلاپی، نوار یا دیسکهای نوری پدید می‌آورد.

سیستم عامل خواص فیزیکی وسایل را از دید کاربر مخفی کرده و یک واحد ذخیره منطقی به نام فایل ارائه می‌کند. سیستم عامل در رابطه با فایل وظایف زیر را انجام می‌دهد:

ایجاد و حذف فایلها

ایجاد و حذف دایرکتوریاها

انجام عملیات کپی

انتقال و تغییرات بر روی فایلها و دایرکتوریاها

ذخیره سازی و مدیریت قرار گیری فایلها بر روی رسانه‌ها

مدیریت دسترسی های مختلف به فایلهای مشترک

مدیریت ورودی- خروجی در سیستم عامل

مفسر فرمان

یکی از مهمترین برنامه‌های سیستم عامل مفسر فرمان است که در واقع واسط بین کاربر و سیستم عامل می‌باشد.

بعضی از سیستم عاملها مفسر فرمان را در هسته خود (kernel) قرار داده‌اند و بعضی دیگر مثل DOS و UNIX مفسر فرمان را (که پوسته یا Shell نیز معروف است) مانند یک برنامه خاص که در اولین برقراری ارتباط اجراء می‌شود در نظر می‌گیرند.

مفسر فرمان دستورات کاربر را گرفته و آنها را اجرا می کند. در پوسته DOS و UNIX فرمانها از طریق صفحه کلید وارد شده و روی صفحه نمایش رایانه به صورت متنی نشان داده می شود ولی در سیستم عامل ویندوز یا مکینتاش پوسته به صورت محیطی گرافیکی و مبتنی بر پنجره هاست که با زدن کلید موس به راحتی می توان دستورات را وارد کرد

وقفه در سیستم عامل

وقفه ها جزء مهمی از معماری کامپیوتر هستند و نحوه عملکرد آنها از ماشینی به ماشین دیگر ممکن است متفاوت باشد. وقفه راهکاری را فراهم می سازد تا اجرای دستورالعملهای جاری پردازنده موقتاً متوقف شده و دستورات سرویس دهی دیگری اجرا گردد و سپس از آن کنترل دوباره به برنامه وقفه داده شده باز گردد.

انواع وقفه ها را می توان به صورت زیر دسته بندی کرد:

وقفه های برنامه (یا program check) که به دلیل اجرای بعضی دستورات رخ می دهند. مثلاً سرریز شدن محاسباتی تقسیم بر صفر، اجراء دستورالعمل غیر مجاز، رجوع به آدرس خرج از محدوده مجاز کاربر. به این وقفه ها اغلب Trap یا تله گفته می شود.

وقفه های زمان سنج (Timer) این وقفه به سیستم عامل امکان می دهد بعضی اعمال را به صورت مرتب در یک پریود زمانی خاص انجام دهد (مثل تنظیم ساعت، چک کردن سخت افزار و..).

وقفه های I/O این وقفه ها به وسیله کنترل کننده های دستگاه I/O تولید می شوند تا کامل شدن طبیعی یک عمل یا بروز خطا در انجام عمل را نشان دهند.

وقفه های نقص سخت افزار یا وقفه های (Machine-check) مثل وقفه ای که بر اثر خطای بیت توازن (parity) حافظه رخ می دهد یا وقفه نقص برق

وقفه (Super Visor Call) که در واقع یک تقاضا از طرف برنامه کاربر جهت دریافت سرویس ویژه ای از سیستم عالم است .

وقفه Restart که با فشار دادن دکمه Reset ایجاد می شود .

در یک تقسیم بندی کلی می توان وقفه های را سه دسته کرد: