



دانشگاه فردوسی مشهد

انتشارات، شماره ۵۲۴

سیستم‌های عامل

(مفاهیم و روش‌ها)

دکتر محمود نقیب زاده

سرشناسه:	نقیب‌زاده، محمود، ۱۳۳۰ -
عنوان و نام پدیدآور:	سیستم‌های عامل (مفاهیم و روش‌ها) / تألیف محمود نقیب‌زاده.
مشخصات نشر:	مشهد: دانشگاه فردوسی مشهد، ۱۳۸۷.
مشخصات ظاهری:	۴۰۰ ص.
فروست:	(دانشگاه فردوسی مشهد؛ شماره ۵۲۴).
شابک:	(ISBN: 978-964-386-191-9)
وضعیت فهرست‌نویسی:	فیا.
موضوع:	سیستم‌های عامل (کامپیوتر).
شناسه افزوده:	دانشگاه فردوسی مشهد
رده‌بندی کنگره:	۱۳۸۷ ن ۷۳ ۹۴ س / ۷۶ / ۷۶ QA
رده‌بندی دیویی:	۰۰۵/۳۴۶
شماره کتابخانه ملی:	۱۵۵۸۹۶۷



دانشگاه فردوسی مشهد

انتشارات، شماره ۵۲۴

سیستم‌های عامل

(مفاهیم و روشها)

تألیف

دکتر محمود نقیب‌زاده

وزیری، ۴۰۰ صفحه، ۱۰۰۰ نسخه، چاپ دوم، پاییز ۱۳۹۰

امور فنی و چاپ: مؤسسه چاپ و انتشارات دانشگاه فردوسی مشهد

بها: ۵۶۰۰۰ ریال

ISBN: 978-964-386-191-9

شابک ۹-۱۹۱-۳۸۶-۹۶۴-۹۷۸

فهرست

۷	پیش گفتار
	فصل اول: سازمان سخت افزار کامپیوتر
۱۶	۱-۱- سیکل واکنشی- اجرا
۲۰	۲-۱- سازمان سخت افزاری کامپیوتر
۲۴	۳-۱- جمع بندی
۲۵	۴-۱- پرسش ها
۲۶	منابع دیگر برای مطالعه بیشتر
	فصل دوم: مراحل بالا آمدن سیستم عامل و نقش BIOS
۲۸	۱-۲- اقدامات BIOS
۲۸	۱-۱-۲- خود آزمایشی روشن شدن
۲۹	۲-۱-۲- نمایش آرم سازنده BIOS
۲۹	۳-۱-۲- اجازه اصلاح اطلاعات CMOS
۳۱	۲-۲- فرآیند بالا آمدن سیستم عامل
۳۸	۳-۲- مکانیزمهای حفاظتی
۴۱	۴-۲- مراحل ساخت یک سیستم عامل کوچک
۴۳	۵-۲- جمع بندی
۴۴	۶-۲- پرسش ها
۴۵	منابع دیگر برای مطالعه بیشتر
	فصل سوم: چند برنامه ای و چند نخه
۴۸	۱-۳- وضعیت های یک فرآیند در تک برنامه ای
۵۰	۲-۳- چند برنامه ای
۵۴	۳-۳- نمودار انتقال حالت فرآیند
۵۶	۴-۳- جایگزینی فرآیند
۵۸	۵-۳- سیستم وقفه
۶۱	۶-۳- دستیابی مستقیم به حافظه
۶۴	۷-۳- چند وظیفه ای
۶۵	۸-۳- چند نخه
۶۷	۹-۳- چند پردازشی

- ۶۸ ۳-۱۰- جمع‌بندی
 ۶۸ ۳-۱۱- پرسش‌ها
 ۷۰ منابع دیگر برای مطالعه بیشتر

فصل چهارم: فرآیند

- ۷۲ ۴-۱- صفات فرآیند
 ۷۸ ۴-۲- فراخوانی‌های سیستم
 ۸۴ ۴-۳- مدل حالتها و انتقال حالت فرآیند در یونیکس
 ۹۲ ۴-۴- ایجاد و از بین رفتن فرآیند
 ۹۳ ۴-۴-۱- چرا فرآیند ایجاد می‌شود؟
 ۹۳ ۴-۴-۲- چه موقع یک فرآیند ایجاد می‌شود؟
 ۹۴ ۴-۴-۳- برای ایجاد یک فرآیند چه کارهایی انجام می‌شود؟
 ۹۵ ۴-۴-۴- چه ابزاری برای ایجاد فرآیند وجود دارد؟
 ۹۸ ۴-۴-۵- چه چیزهایی در فرآیند پدر و فرزند یکسان است؟
 ۹۹ ۴-۴-۶- چه چیزهایی بین فرآیندهای پدر و فرزند متفاوت است؟
 ۱۰۰ ۴-۴-۷- چگونه یک فرآیند پایان می‌پذیرد؟
 ۱۰۰ ۴-۵- سیستم عامل مبتنی بر فرآیند
 ۱۰۱ ۴-۶- جمع‌بندی
 ۱۰۲ ۴-۷- پرسش‌ها
 ۱۰۳ منابع دیگر برای مطالعه بیشتر

فصل پنجم: نخ

- ۱۰۷ ۵-۱- انواع نخ
 ۱۱۰ ۵-۲- بخشهای مشترک نخ‌های خواهر- برادر
 ۱۱۲ ۵-۳- صفات نخ
 ۱۱۵ ۵-۴- حالتها و انتقال حالتها در نخ درویندوز
 ۱۱۶ ۵-۵- ایجاد و پایان نخ
 ۱۱۷ ۵-۵-۱- چرا نخ ایجاد می‌شود؟
 ۱۱۸ ۵-۵-۲- چه موقع یک نخ ایجاد می‌شود؟
 ۱۱۹ ۵-۵-۳- برای ایجاد یک نخ چه کارهایی انجام می‌شود؟
 ۱۱۹ ۵-۵-۴- چه ابزاری برای ایجاد نخ وجود دارد؟
 ۱۲۳ ۵-۵-۵- چگونه نخ پایان می‌پذیرد؟
 ۱۲۴ ۵-۵-۶- پس از پایان نخ چه کارهایی انجام می‌شود.
 ۱۲۴ ۵-۶- سیستم‌های عامل مبتنی بر نخ

۱۲۵	۷-۵- جمع‌بندی
۱۲۵	۸-۵- پرسش‌ها
۱۲۷	منابع دیگر برای مطالعه بیشتر
	فصل ششم: زمانبندی
۱۳۰	۱-۶- معیارهای زمانبندی
۱۳۰	۱-۱-۶- زمان درخواست
۱۳۰	۲-۱-۶- مدت پردازش
۱۳۱	۳-۱-۶- اولویت
۱۳۱	۴-۱-۶- ضرب الاجل
۱۳۲	۵-۱-۶- مدت انتظار
۱۳۲	۶-۱-۶- قبضه‌پذیری
۱۳۲	۲-۶- اهداف زمانبندی
۱۳۳	۱-۲-۶- برون‌دهی
۱۳۴	۲-۲-۶- ضرب الاجل
۱۳۴	۳-۲-۶- مدت پاسخگویی
۱۳۶	۴-۲-۶- مدت تا عکس‌العمل
۱۳۶	۵-۲-۶- اولویت
۱۳۷	۶-۳-۶- کاربری پردازشگر
۱۳۷	۷-۲-۶- تراز کردن سیستم
۱۳۷	۸-۲-۶- رعایت عدالت
۱۳۸	۳-۶- سطوح زمانبندی
۱۳۹	۱-۳-۶- زمانبندی سطح بالا
۱۳۹	۲-۳-۶- زمانبندی سطح میانی
۱۴۰	۳-۳-۶- زمانبندی سطح پایین
۱۴۱	۴-۳-۶- زمانبندی ورود و خروج
۱۴۴	۴-۶- الگوریتم‌های زمانبندی سیستم‌های یک پردازنده
۱۴۸	۱-۴-۶- اولین وارده - اولین سرویس گیرنده
۱۵۰	۲-۴-۶- ابتدا کوتاهترین کار
۱۵۱	۳-۴-۶- ابتدا فرآیند با کمترین زمان اجرای باقیمانده
۱۵۲	۴-۴-۶- ابتدا فرآیند با بیشترین نسبت پاسخگویی
۱۵۴	۵-۴-۶- زمانبندی سهم عادلانه
۱۵۴	۶-۴-۶- نوبت چرخشی
۱۵۸	۷-۴-۶- زمانبندی اولویتی

۶ سیستم‌های عامل (مفاهیم و روش‌ها)

- ۱۵۸ ۶-۴-۸- زمانبندی فرآیندها در لینوکس
- ۱۶۲ ۶-۵- تحلیل الگوریتم‌های زمانبندی
- ۱۶۵ ۶-۵-۱- متوسط زمان عکس العمل
- ۱۶۸ ۶-۶- زمانبندی در چند پردازنده‌ها
- ۱۶۸ ۶-۶-۱- نوع پردازشگرها
- ۱۶۸ ۶-۶-۲- تمایل به پردازشگر
- ۱۷۰ ۶-۶-۳- بسامد هماهنگ سازی
- ۱۷۱ ۶-۶-۴- انتساب ایستا در برابر انتساب پویا
- ۱۷۲ ۶-۷- الگوریتم‌های زمانبندی در سیستم‌های چند پردازنده
- ۱۷۲ ۶-۷-۱- اولین وارده اولین سرویس گیرنده
- ۱۷۳ ۶-۷-۲- ابتدا کوتاهترین کار
- ۱۷۳ ۶-۷-۳- ابتدا کوتاهترین زمان باقیمانده
- ۱۷۴ ۶-۷-۴- زمانبندی سهم عادلانه
- ۱۷۴ ۶-۷-۵- زمانبندی نوبت چرخشی
- ۱۷۵ ۶-۷-۶- زمانبندی تیمی
- ۱۷۶ ۶-۷-۷- زمانبندی SMP در لینوکس
- ۱۷۷ ۶-۸- زمانبندی فرآیندهای بی درنگ
- ۱۷۹ ۶-۸-۱- زمانبندی متناسب با بسامد درخواست
- ۱۸۱ ۶-۸-۲- رهیافت ابتدا نزدیکترین ضرب الاجل
- ۱۸۳ ۶-۸-۳- رهیافت ابتدا کمترین فرجه تا آغاز
- ۱۸۵ ۶-۹- زمانبندی خواندن/نوشتن دیسک
- ۱۹۰ ۶-۹-۱- الگوریتم اولین وارده - اولین خارجه
- ۱۹۲ ۶-۹-۲- الگوریتم آخرین وارده - اولین خارجه
- ۱۹۲ ۶-۹-۳- الگوریتم ابتدا کوتاهترین مدت خدمت رسانی
- ۱۹۳ ۶-۹-۴- الگوریتم پویش
- ۱۹۵ ۶-۹-۵- الگوریتم پویش حلقوی
- ۱۹۶ ۶-۱۰- جمعبندی
- ۱۹۸ ۶-۱۱- پرسش‌ها
- ۲۰۳ منابع دیگر برای مطالعه بیشتر

فصل هفتم: مدیریت حافظه اصلی

- ۲۰۸ ۷-۱- سیاست‌های قدیمی مدیریت حافظه اصلی
- ۲۰۸ ۷-۱-۱- مدیریت حافظه به روش یک قطعه یکجا
- ۲۰۹ ۷-۱-۲- مدیریت حافظه با افراز ایستا

فهرست ۷

۲۱۱	۳-۱-۷- مدیریت حافظه با افراز پویا
۲۱۹	۴-۱-۷- مدیریت حافظه چند قطعه‌ای
۲۲۰	۲-۷- سیاست‌های مدیریت حافظه با کاربردهای محدود
۲۲۱	۱-۲-۷- مدیریت حافظه چند بخشی
۲۲۴	۲-۲-۷- مدیریت حافظه قطعه بندی جا به جا پذیر
۲۲۶	۳-۲-۷- مدیریت حافظه صفحه بندی
۲۳۴	۳-۷- حافظه مجازی
۲۳۵	۴-۷- مدیریت حافظه مجازی مبتنی بر صفحه
۲۴۳	۱-۴-۷- تبدیل آدرس
۲۵۰	۲-۴-۷- الگوریتم‌های اخراج صفحه
۲۶۹	۳-۴-۷- اندازه بهینه صفحه
۲۷۰	۴-۴-۷- حافظه مجازی مبتنی بر جدول صفحات دوسطحی
۲۷۲	۵-۷- حافظه مجازی با قطعه بندی و صفحه بندی
۲۷۴	۶-۷- مدیریت حافظه انبارک
۲۸۱	۷-۷- مدیریت حافظه اصلی در سیستم عامل ویندوز
۲۸۶	۸-۷- مدیریت حافظه اصلی در یونیکس
۲۸۸	۹-۷- جمع‌بندی
۲۹۰	۱۰-۷- پرسش‌ها
۲۹۷	منابع دیگر برای مطالعه بیشتر

فصل هشتم: هماهنگی و همزمانی فرآیندها

۳۰۰	۱-۸- دلایل هماهنگی سازی و همزمان سازی فرآیندها
۳۰۰	۱-۱-۸- شرایط مسابقه
۳۰۶	۲-۱-۸- منطقه بحرانی
۳۰۷	۳-۱-۸- مساله یا این یا آن
۳۰۹	۲-۸- فناوری‌های هماهنگی و همزمانی فرآیندها
۳۱۱	۱-۲-۸- روش از کار انداختن سیستم وقفه
۳۱۲	۲-۲-۸- روشهای IPC مبتنی بر انتظار-مشغول
۳۲۳	۳-۲-۸- هماهنگی و همزمانی مبتنی بر علامت ده
۳۳۰	۴-۲-۸- هماهنگی و همزمانی مبتنی بر Monitor
۳۳۲	۵-۲-۸- هماهنگی و همزمانی مبتنی بر پیام
۳۳۶	۳-۸- جمع‌بندی
۳۳۷	۴-۸- پرسش‌ها
۳۴۲	منابع دیگر برای مطالعه بیشتر

فصل نهم: بن بست و تلف شدن

۳۴۳	۱-۹- بن بست
۳۴۴	۱-۱-۹ شرایط بن بست
۳۴۸	۲-۱-۹ نادیده گرفتن بن بست
۳۴۹	۳-۱-۹ پیشگیری از بن بست
۳۵۴	۴-۱-۹ پرهیز از بن بست
۳۶۳	۵-۱-۹ تشخیص و ترمیم بن بست
۳۶۹	۲-۹ تلف شدن
۳۷۰	۳-۹ یک مساله IPC کلاسیک
۳۷۷	۴-۹ جمع بندی
۳۷۸	۵-۹ پرسش‌ها
۳۸۵	منابع دیگر برای مطالعه بیشتر
۳۸۷	مراجع
۳۹۳	نمایه

Press.um.ac.ir

به نام خداوند جان و خرد

پیش‌گفتار

انسان امروزی برای انجام بیشتر فعالیت‌های پیشرفته خود از کامپیوتر بهره می‌برد. استفاده از رایانه به سن خاصی محدود نمی‌گردد و توسط پیر و جوان و زن و مرد و در محیط‌های گوناگون به کار گرفته می‌شود. تصور عامیانه از یک کامپیوتر مجموعه‌ای از دستگاهها مانند، صفحه نمایش، صفحه کلید، موش، جعبه اصلی و... است که یکجا سخت افزار نامیده می‌شود. در یک جلسه کار با کامپیوتر ابتدا آن را روشن کرده چند ثانیه صبر می‌کنیم که آماده شود، و سپس از سیستم عامل می‌خواهیم برنامه‌ای را برای ما اجرا کند. لذا این یک دیدگاه درست است که هر کامپیوتر یک سیستم عامل دارد.

سیستم عامل مهم ترین برنامه در کامپیوتر است و بدون وجود آن کار با کامپیوتر سخت و ملال آور است. این نرم افزار رابطی بین سخت افزار و کاربر است که کار با کامپیوتر را ساده تر و لذت بخش می‌کند. آشنا شدن با سیستم عامل استفاده از کامپیوتر را ساده تر می‌کند و کمک می‌کند که کاراتر از امکانات آن استفاده کنیم و کارهای خود را به خوبی انجام دهیم. سیستم عامل به گونه‌ای طراحی و پیاده سازی می‌گردد که از سخت افزار به شکل مطلوب و با بهره‌وری بالا در جهت اجرای درخواستهای کاربران استفاده کند. لذا بهتر است انواع خدمات قابل ارائه توسط کامپیوتر و نحوه استفاده از این خدمات را بدانیم. سیستم عامل باید به گونه‌ای طراحی گردد که از سخت افزار به بهترین شکل در جهت اجرای درخواستهای کاربران استفاده کند. در این راستا تلاش فراوانی شده است. از بدو اختراع کامپیوتر در دهه ۱۹۴۰ متخصصان همت زیادی صرف روان سازی کار با این نرم افزار از یک طرف و بالا بردن کارایی آن از طرف دیگر کرده‌اند. حاصل این تلاشها شاهکاری است که ثمره سخت کوشیها و نوآوریهای فراوان را یکجا جمع کرده است.

ده‌ها سیستم عامل با اندازه‌ها و نام‌های مختلف و صدها کتاب و منبع علمی برای بیان روش استفاده از آنها وجود دارد. از آن گذشته مؤلفان زیادی منابعی را در ارتباط با تشریح اصول طراحی سیستم عامل نوشته‌اند. تالیف‌های از این دسته توسط متخصصان مرتبط و دانشجویان دانشگاهها که در این زمینه درس انتخاب می‌کنند، مورد استفاده قرار می‌گیرد.

در این کتاب، ابتدا راجع به روش تسلط سیستم عامل بر سخت افزار و مالک شدن آن، که کمتر صحبت شده است، بحث خواهد شد. این دانش به همراه یک دیدگاه کلی از مسئولیت‌های سیستم عامل و روش انجام این مسئولیت‌ها امتیازی است که فرد را برای استفاده بهتر از امکانات کامپیوتر آماده می‌کند. از آن گذشته، می‌تواند زمینه خوبی برای درک بهتر مفاهیم علمی آن باشد. در ادامه به بیان مفاهیم، روشها و فنون به کار رفته در طراحی سیستم‌های عامل پرداخته شده است.

این کتاب برای همه کسانی است که از کامپیوتر استفاده می‌کنند اما با آن راحت نیستند و نمی‌دانند چگونه سیستم عامل منابع را به بهترین شکل به کار می‌گیرد که درخواست‌های کاربران را صحیح و سریع انجام دهد. دانشجویان به این دلیل که برای بیشتر کارهای آموزشی، پژوهشی، اینترنت، و حتی تفریح خود کامپیوتر را به کار می‌گیرند از این کتاب بیشترین استفاده را خواهند برد. آنها عمدتاً کنجکاو هستند و می‌خواهند درباره نرم افزاری که پشت پرده کمک می‌کند همه چیز یک جریان روان داشته باشد، بیشتر بدانند. بویژه دانشجویان تکنولوژی اطلاعات و علوم و مهندسی کامپیوتر بیشترین بهره را خواهند برد. به نظر من هر دانشجو که درس دانشگاهی سیستم عامل را می‌گیرد باید این کتاب را بخواند. این کتاب شرح دقیقی از مفاهیم پیشرفته زمینه را به همراه روش بکارگیری آن در سیستم‌های عامل مطرح روز تدوین کرده است. حجم کتاب با حذف برخی از مباحث طولانی که تنها جنبه تاریخی دارد و دانش روز محسوب نشده و کاربرد جاری ندارد به حداقل کاهش یافته است.

در فصل اول، یک مدل ساده از سخت افزار کامپیوتر تدوین شده است. در این مدل از هرگونه جزئیاتی که به مباحث اصلی کتاب مربوط نمی‌گردد اجتناب شده است. تمرکز اصلی این فصل بر تشریح سیکل واکنشی و اجرای دستورات است. بر ریزه کاریهای این عملیات که در فصل‌های بعدی به آنها نیاز خواهد بود تاکید گردیده است.

در فصل دوم، نقش نرم افزار عملیات ورود و خروج سیستم در واریسی اولیه سخت افزار و روش بالا آمدن سیستم عامل را بررسی کرده‌ایم. ساختار سلسله مراتبی در پیاده سازی سیستم عامل مورد توجه قرار گرفته و در مورد هسته سیستم عامل و نقش آن در کمک رسانی به نیازهای لایه‌های بالاتر سیستم عامل صحبت شده است. حراست و حفاظت به عنوان دو موضوع

اساسی نیز مورد توجه قرار گرفته است.

فصل سوم، لزوم استفاده از چند برنامه‌ای، چند وظیفه‌ای، چند پردازشی و چندنخی را بیان می‌کند. شرح مختصری از حالت‌های فرآیند، نمودار انتقال از حالتی به حالت دیگر، پرش از فرآیندی به فرآیند دیگر، سیستم وقفه، و سیستم دستیابی مستقیم به حافظه اصلی در این فصل آمده است.

فصل چهارم به فرآیند اختصاص دارد. به سوالهای زیادی پاسخ داده شده است. فرآیند چیست؟، چرا و چگونه ساخته می‌شود؟، چه اطلاعاتی برای مدیریت فرآیندها لازم است؟، چگونه یک فرآیند پایان می‌پذیرد؟، رابطه بین فرآیند پدر و فرآیند فرزند چیست؟ و چه زمانی فرآیند تمام می‌شود؟، برخی از این سوالها هستند. گفتار کوتاهی راجع به ویژگیهای سیستم‌های عامل مبتنی بر فرآیند به اختصار به بحث گذاشته شده است. مثالهای عملی این فصل را از سیستم عامل یونیکس که یک سیستم عامل مبتنی بر فرآیند است آورده‌ایم. یکی از موضوع‌های مهم تعیین وضعیت‌های یک فرآیند و نقل و انتقال‌های ممکن بین وضعیت‌ها است که بخش مهمی از فصل را تشکیل می‌دهد.

فصل پنجم به نخ اختصاص داده شده است. باوجود فرآیند، نیاز به نخ و رابطه این دو را روشن کرده‌ایم. برخی از سیستم‌های عامل مبتنی بر نخ هستند. توابع هسته اینگونه سیستم‌های عامل توسط نخ انجام می‌شود. چرا و چگونه نخ تولید می‌شود؟ به چه اطلاعاتی برای مدیریت نخ‌ها نیاز است؟ یک نخ چگونه پایان می‌پذیرد؟ و رابطه بین نخ پدر و نخ فرزند چیست در این فصل بررسی شده است. حالت‌های نخ‌ها را بررسی کرده‌ایم و نقل و انتقال‌های ممکن بین این حالتها به بحث گذاشته شده است. مثال‌های این فصل را از سیستم عامل ویندوز آورده‌ایم که یک سیستم عامل مبتنی بر نخ است.

زمانبندی موضوع فصل ششم است. زمانبندی یکی از مفاهیم تاثیرگذار بر طراحی سیستم‌های عامل است. در ابتدا واژه‌های کلیدی زمانبندی مانند زمان درخواست، زمان پردازش، اولویت، ضرب الاجل، مدت انتظار و قبضه‌پذیری را تعریف و تشریح کرده‌ایم. در ادامه اهداف زمانبندی مانند بیشینه کردن برون‌دهی، رعایت ضرب الاجل، کاهش متوسط مدت پاسخگویی، بیشینه کردن زمان به کارگیری پردازشگر، کاهش متوسط مدت پاسخ اولیه، رعایت اولویت، تراز کردن سیستم و رعایت عدالت را شرح دادیم. چهار سطح زمانبندی شامل سطح بالا، سطح متوسط، سطح پایین و زمانبندی ورود و خروج تشخیص داده شده‌اند. زمانبندی‌های زیادی که در سیستم‌های تک پردازشگر کاربرد دارند را به دقت مطالعه کرده‌ایم. این زمانبندیها از جمله خدمات به ترتیب ورود، ابتدا کوتاهترین درخواست، ابتدا کوتاهترین باقیمانده درخواست، ابتدا بالاترین

نسبت پاسخگویی، رعایت عادلانه سهم فرآیندها، و نوبت چرخشی هستند. بیشتر این روش‌های زمانبندی برای محیط‌های چند پردازشگر نیز کاربرد دارند. افزون بر آنها، زمانبندی گروهی نیز برای محیط‌ها چند پردازشگر بررسی شده است. برای محیط‌های بلادرنگ زمانبندی متناسب با بسامد درخواست و ابتدا نزدیکترین ضرب الاجل کاربرد وسیعی یافته‌اند. این زمانبندها در این فصل معرفی شده‌اند. مثال‌های واقعی از زمانبندها از سیستم عامل لینوکس گرفته شده است.

موضوع فصل هفتم کتاب سیاست‌های مدیریت حافظه اصلی است. سیاست‌های قدیمی مانند مدیریت حافظه به روش یک قطعه یکجا، افراز ایستا، افراز پویا و قطعه بندی به اختصار توضیح داده شده است. در ادامه مدیریت‌های حافظه غیرمجازی، مدیریت حافظه با افراز جا به جاپذیر و مدیریت حافظه صفحه بندی شده را مطالعه کرده‌ایم. مدیریت حافظه مجازی مبتنی بر صفحه مهمترین سیاست مدیریتی روز است که بوسیله بیشتر سیستم‌های عامل مدرن به کار گرفته شده است. بخش بزرگی از این فصل به این سیاست اختصاص دارد. به مساله روش بیرون کردن صفحه از حافظه اصلی اهمیت ویژه‌ای داده شده است. جدول زمانبندی برنامه‌های بزرگ جای زیادی در حافظه اصلی اشغال می‌کند. جدول صفحات چند سطحی راهی برای تعمیم فناوری حافظه مجازی به جدول صفحات است. این فناوری را مطالعه کرده‌ایم. ویندوز از مدل دو سطحی جدول صفحات استفاده می‌کند و ما آن را با جزئیات مطالعه خواهیم کرد. این مطالعه کاربرد عملی سیاست‌های مدیریت حافظه اصلی را در یک سیستم عامل مدرن ارزیابی می‌کند. تبدیل آدرس منطقی به آدرس فیزیکی از شاخصهای مدیریت‌های حافظه مبتنی بر صفحه، اعم از حقیقی یا مجازی، است که بحث نسبتاً مشکلی است؛ جزئیات عملیات را به دقت باز کرده‌ایم. تکه تکه شدن داخلی و خارجی باعث هدر رفتن حافظه اصلی می‌گردد و سیستم‌های عامل سعی می‌کنند در راستای کاهش آن گام بردارند. روش‌های محاسبه مقدار حافظه تلف شده در هر یک از مدیریت‌های حافظه اصلی در این فصل بررسی شده است. حافظه انبارک در بالا بردن کارایی سیستم‌ها نقش ارزنده‌ای دارد. در انتهای این فصل روش‌های مدیریت این نوع حافظه را مطالعه کرده‌ایم. مثال‌های واقعی این فصل از سیستم‌های عامل یونیکس و ویندوز آمده‌اند.

فرآیندهایی که همزمان در کامپیوتر وجود دارند برای گرفتن هرچه بیشتر و سریعتر منابع مورد نیاز باهم رقابت می‌کنند. تضاد منافع می‌تواند منجر به تولید نتایج نادرست توسط برنامه‌های درست گردد. داوری و حل تضادها نیازمند حضور یک میانجی است. این میانجی زیر سیستم ارتباطات/هماهنگی بین فرآیندی است. ابتدا شرایط مسابقه، منطقه بحرانی، مساله "یا این یا آن" در فصل هشت تشریح شده‌اند. نشان داده شده است که با حل مساله یا این یا آن استفاده اشتراکی از منابع بدون این که فرآیندها نتیجه نادرست تولید کنند امکانپذیر است. روش‌های حل

مساله یا این یا آن به پنج دسته تقسیم می‌شوند. نادیده گرفتن وقفه، مبتنی بر انتظار-مشغول، مبتنی بر سمافر، مبتنی بر مانیتور، و مبتنی بر پیام این پنج دسته هستند که همگی در فصل ۸ تشریح شده‌اند.

بن بست از آثار ناپسند حل مساله یا این یا آن است. در فصل ۹ این مساله بررسی شده و راه‌های برخورد با آن مطالعه شده است. نادیده گرفتن بن بست، پیش‌گیری از بن بست، پرهیز از بن بست، و تشخیص و رفع بن بست چهار رهیافت برخورد با مساله بن بست است. یک مساله کم‌اهمیت‌تر از بن بست مساله تلف شدن از گرسنگی است که آنهم در این فصل شناسایی و مطالعه شده. مساله شام خوردن فیلسوفها به عنوان یک مساله جا افتاده که دادن راه حل بدون بن بست و گرسنگی برای آن بدیهی نیست بررسی شده است. نشان داده‌ام که راه حل‌های ساده و روان دارای مشکل بن بست و یا احتمال گرسنگی است. یک راه حل فاقد بن بست و فاقد احتمال گرسنگی را در ادامه ارائه کرده‌ام.

این کتاب را به همسر و فرزندانم تقدیم می‌کنم.

دکتر محمود نقیب‌زاده
۱۳۸۲ شمسی (۲۰۰۹ میلادی)