

هو برنامج يُعتبر (OS) تعريف نظام التشغيل (Introduction & What Operating Systems Do) مقدمة عن نظم التشغيل 1. والمستخدمين، ويهدف إلى ضمان تنفيذ البرامج بكفاءة وسهولة. (Hardware) بين عتاد الحاسوب (Interface) بمثابة الوسيط "إدارة موارد الحاسوب (مثل: المعالج، الذاكرة، وحدات التخزين، وأجهزة الإدخال والإخراج). العمل كـ "موزع للموارد ومتحكم في البرامج لمنع الأخطاء وسوء استخدام الجهاز. تلبية احتياجات المستخدمين (السهولة (Resource Allocator) وسرعة الأداء في الأنظمة الشخصية، مقابل كفاءة واستغلال الموارد في الأنظمة المشتركة (الخوادم). 2. مكونات نظام الحاسوب الموارد: (Hardware) ينقسم نظام الحاسوب إلى 4 مكونات رئيسية: 1. العتاد (Computer System Components) يتحكم: (Operating System) الذاكرة، وأجهزة الإدخال والإخراج. 2. نظام التشغيل، CPU الأساسية للحاسوب (المعالج تحدد كيفية: (Application Programs) وينسق استخدام العتاد بين التطبيقات والمستخدمين المختلفين. 3. برامج التطبيقات استخدام موارد النظام لحل مشكلات المستخدم (مثل: متصفحات الويب، معالجات النصوص، الألعاب). 4. المستخدمون (Computer) الأشخاص، الآلات، أو الأنظمة الأخرى التي تستخدم الحاسوب. 3. تنظيم نظام الحاسوب والعمليات (Users) آلية العمل: يتكون النظام من معالج (أو عدة معالجات) ومتحكمات أجهزة تتصل فيما (System Organization & Operation) تعتمد نظم التشغيل بشكل (Interrupts) للوصول إلى الذاكرة المشتركة. المقاطعات (Common Bus) بينها عبر ناقل مشترك أساسي على المقاطعات؛ حيث يقوم جهاز الإدخال/الإخراج بإرسال إشارة مقاطعة للمعالج عند إتمام مهمته لتغيير مسار التنفيذ لحماية النظام، وضع: (Dual-Mode Operation) الوضع المزدوج. (Interrupt Vector) مؤقتاً عبر جدول ناقل المقاطعات للعمليات الحساسة والمصرح لها: (Kernel Mode) لتنفيذ تطبيقات البرامج العادية. وضع النواة: (User Mode) المستخدم (Process, Memory, إدارة العمليات والذاكرة والتخزين 4. (Mode Bit) بصلاحيات كاملة، ويتم التبديل بينهما عبر بت نمط وتعتبر وحدة العمل (Program in execution) العملية هي برنامج قيد التنفيذ: (Process Management) إدارة العمليات الأساسية في النظام. يتولى نظام التشغيل مسؤوليات مثل إنشاء/حذف العمليات، إيقافها واستئنافها، وتوفير آليات المزامنة تتطلب تنفيذ البرامج تحميل تعليماتها: (Memory Management) إدارة الذاكرة. (Deadlocks) والاتصال ومعالجة التجاوزات وبياناتها (أو جزء منها) إلى الذاكرة الرئيسية. يتابع نظام التشغيل الأجزاء المستعملة من الذاكرة ومن يستخدمها، ويخصص أو يوفر نظام التشغيل نظرة منطقية موحدة: (Storage Management) يحرر مساحات الذاكرة حسب الحاجة. إدارة التخزين (Directories) وإدارتها في مجلدات (Files) "لتخزين المعلومات عبر تجريد الخصائص الفيزيائية لوحدة التخزين إلى "ملفات أنظمة المعالجات المتعددة (System Architectures & Open-Source OS) المعماريات والأنظمة مفتوحة المصدر 5. (Symmetric وتنقسم إلى معالجة متماثلة، (Fault Tolerance) تزيد من إنتاجية النظام وموثوقيته: (Multiprocessors) (Open-Source نظم التشغيل مفتوحة المصدر. (Asymmetric Multiprocessing) وغير متماثلة (Operating Systems) و Linux تتيح الشيفرة البرمجية للمستخدمين ليتمكنوا من دراستها، تعديلها، وتحسينها (مثل أنظمة: BSD UNIX) يقدم نظام التشغيل بيئة لتنفيذ البرامج وخدمات (Operating System Services) خدمات نظم التشغيل 1. (User Interface - UI) للمستخدمين، واجهة المستخدم توفر وسيلة للتفاعل مع المستخدم، وتكون إما عبر سطر الأوامر: (Batch) أو الأوامر النصية، (GUI) أو الواجهة الرسومية، (CLI) قدرة النظام على: (Program Execution) تنفيذ البرامج. (I/O تحميل البرنامج إلى الذاكرة وتنفيذه وإنهائه (سواء بشكل طبيعي أو في حال حدوث خطأ). عمليات الإدخال والإخراج (File-system التعامل مع طلبات البرامج المتعلقة بملفات أو أجهزة الإدخال والإخراج. معالجة وملفات النظام: (Operations قراءة، كتابة، إنشاء، حذف، والبحث في الملفات والمجلدات مع إدارة الصلاحيات. الاتصالات: (Manipulation تبادل المعلومات بين العمليات سواء على نفس الحاسوب أو عبر الشبكة (عن طريق الذاكرة المشتركة: (Communications مراقبة الأخطاء في: (Error Detection) اكتشاف الأخطاء. (Message Passing أو تمرير الرسائل Shared Memory Resource Allocation): (المعالج، الذاكرة، أو الأجهزة واتخاذ الإجراء المناسب لضمان استقرار العمل. تخصيص الموارد (Accounting) توزيع الموارد (مثل دورات المعالج، الذاكرة، والتخزين) عند تشغيل عدة مستخدمين أو مهام بالتوازي. المحاسبة التحكم في الوصول إلى موارد: (Protection and Security) تتبع استخدام المستخدمين للموارد وكميتها. الحماية والأمان النظام، وتوثيق هوية المستخدمين وحماية الأجهزة الخارجية من محاولات الوصول غير المرغوب فيها. 2. استدعاءات النظام للخدمات التي يقدمها نظام التشغيل، وتُكتب عادة (Programming Interface) التعريف: هي واجهة البرمجة (System Calls)

Win32 مثل (API) آلية العمل: يتم الوصول إليها غالباً عبر واجهة برمجة التطبيقات C++ أو C بلغات عالية المستوى مثل هذه الأرقام (Table) لأنظمة يونكس ولينكس. يتم تخصيص رقم لكل استدعاء نظام، ويحتفظ الجدول POSIX لأنظمة ويندوز أو مع إخفاء تفاصيل التنفيذ المعقدة عن المبرمج. أنواع استدعاءات النظام الرئيسية: 1. التحكم (Kernel) لتنفيذ الطلب داخل النواة (File) مثل إنشاء/إنهاء العمليات، تحميل وتلبس البرامج، وإدارة الأخطاء. 2. إدارة الملفات: (Process Control) بالعمليات طلب أو تحرير: (Device Management) إنشاء، حذف، فتح، قراءة، وكتابة الملفات. 3. إدارة الأجهزة: (Information Maintenance) التاريخ، (Communications) إنشاء وحذف روابط الاتصال وإرسال واستقبال الرسائل. 6. (System Programs) التحكم في الوصول للموارد والصلاحيات. 3. برامج النظام: (Protection) الحماية توفر بيئة ملائمة (Compilers) محرر النصوص والأوامر الخاصة بالبحث والتعديل. دعم لغات البرمجة: المترجمات: (Modification) ومحرري الروابط (Loaders) تحميل وتنفيذ البرامج: برامج التحميل (Debuggers) والمصححات (Assemblers) المجمعات برامج تعمل في الخلفية منذ الإقلاع لتقديم خدمات: (Background Services / Daemons) الخلفيات (Linkeditors) الأهداف: تحديد (OS Design and Implementation) مثل جدولة العمليات وفحص الأقراص. 4. تصميم وتنفيذ نظم التشغيل أهداف النظام (أهداف المستخدم لتكون سهلة وسريعة، وأهداف النظام لتكون سهلة الصيانة ومرنة). فصل السياسة عن الآلية (How to do it?) الآلية: تحديد "كيف سيتم فعله؟". (What will be done?) "السياسة: تحديد "ماذا سيفعل؟". (Policy vs. وغالباً ما C++ و C ثم لغات مثل (Assembly) فصلهم يمنح مرونة عالية للنظام. لغات التنفيذ: تبدأ الأنظمة عادة بلغة التجميع (Process Concept) تستخدم مزيجاً من اللغات لتحقيق السرعة والكفاءة وسهولة النقل للأجهزة المختلفة. 1. مفهوم العمليات وتعتبر أساس كل عمليات الحاسوب في نظام التشغيل. (Program in Execution) التعريف: العملية هي برنامج قيد التنفيذ كيان (Process) مخزن على القرص الصلب (ملف تنفيذي). العملية (Passive entity) كيان خامل (Program) البرنامج كود البرنامج. العداد (Text section) يتم إنشاؤه عند تحميل البرنامج إلى الذاكرة وتنفيذه. النص (Active entity) نشط يحتوي على البيانات المؤقتة (Stack) لتتبع النشاط الحالي. المكس (Program Counter & CPU Registers) والمسجلات يحتوي على المتغيرات العامة: (Data section) (مثل معلمات الدوال، عناوين العودة، والمتغيرات المحلية). قسم البيانات حالات العملية. 2. (Run time) الذاكرة المخصصة ديناميكياً أثناء وقت التشغيل: (Heap) الكومة. (Global variables) يتم تنفيذ تعليمات العملية بواسطة المعالج. (Running) العملية قيد الإنشاء. قيد التنفيذ: (New) جديدة (Process States) تنتظر العملية لتُعين: (Ready) تنتظر العملية حدوث حدث معين (مثل اكتمال إدخال/إخراج). جاهزة: (Waiting) منتظرة (Process Control Block - انتهت العملية من التنفيذ. 3. كتلة تحكم العملية: (Terminated) وتُخصص لمعالج. منتهية (Process number / رقم العملية). (Process state) هو هيكل بيانات في نظام التشغيل يمثل كل عملية، حالة العملية (PCB) ومعلومات الجدولة وإدارة الذاكرة. 4. جدولة (CPU registers) مسجلات المعالج. (Program counter) عداد البرنامج. (PID) والتبديل السريع بين العمليات لتقاسم (CPU utilization) الهدف: تعظيم استفادة المعالج (Process Scheduling) العمليات يحتوي على جميع: (Job queue) طابور الوظائف: (Scheduling Queues) طوابير الجدولة. (Time sharing) الوقت العمليات المتواجدة في الذاكرة الرئيسية وجاهزة للتنفيذ. طوابير: (Ready queue) العمليات في النظام. طابور الجاهزية (Short-term scheduler / العمليات المنتظرة لأجهزة الإدخال والإخراج. الجدول قصير الأجل: (Device queues) الأجهزة يختار العمليات الجاهزة لتنفيذها بالمعالج (يُستدعى بشكل متكرر وسريع جداً بالمللي ثانية). الجدول طويل (CPU scheduler) يختار العمليات لجلبها إلى طابور الجاهزية (يُستدعى ببطء أكثر، بالثواني: (Long-term scheduler / Job scheduler) الأجل تقضي وقتاً في عمليات الإدخال والإخراج أكثر من الحسابات ولها نوبات قصيرة بالمعالج: I/O-bound process. (أو الدقائق تقضي وقتاً طويلاً في العمليات الحسابية والمعالجة. 5. إنشاء العمليات والعمليات المتزامنة: CPU-bound process (Child) بإنشاء عمليات فرعية (Parent process) الإنشاء: تقوم العملية الرئيسية (Process Creation & Concurrency)

خيارات مشاركة الموارد: تتشارك العمليات (PID) لتشكل شجرة من العمليات يتم تعريفها عبر معرف العملية (processes)، الأب والأبناء إما في جميع الموارد، أو جزء منها، أو لا تتشارك شيئاً. خيارات التنفيذ: قد تنفذ العمليات بالتزامن (Interprocess Communication - Concurrently) أو ينتظر الأب حتى تنتهي العمليات الفرعية. 6. الاتصال بين العمليات (Shared memory) العمليات قد تكون مستقلة أو متعاون، وتتم عملية الاتصال والتبادل بينها إما عن طريق الذاكرة المشتركة (IPC) هو (Thread) التعريف: الخيط (Thread Concept) مفهوم الخيوط 1. (Message passing) أو تمرير الرسائل (memory Stack). ويتكون من عداد برنامج، ومجموعة مسجلات، ومكدس، (CPU utilization) الوحدة الأساسية لاستخدام المعالج داخل العملية الواحدة (Multithreading) العلاقة بالعملية: الخيط يقع داخل العملية، وتدعم المعالجات الحديثة تعدد الخيوط لتنفيذ مهام متعددة في نفس الوقت (مثل تحديث الشاشة، جلب البيانات، والتدقيق الإملائي في نفس التطبيق). 2. فوائد تعدد السماح بتمرير التنفيذ حتى لو كان جزء من العملية: الاستجابة (Responsiveness) (Benefits of Multithreading) الخيوط تتشارك الخيوط موارد العملية التابعة لها (الذاكرة والملفات): (Resource Sharing) محظوراً أو مشغولاً. مشاركة الموارد (Context) إنشاء الخيوط وتبديل السياق: (Economy) بسهولة أكبر مقارنة بمشاركة الذاكرة أو تمرير الرسائل. الاقتصاد استغلال بني المعالجات متعددة النوى: (Scalability) أسرع وأقل تكلفة من إنشاء العمليات. قابلية التوسع (switching) تفرض الأنظمة (Multicore Programming) بشكل أفضل. 3. البرمجة متعددة النوى (Multiprocessor architectures) واختبار وتصحيح (Data dependency) متعددة النوى تحديات في البرمجة مثل تقسيم الأنشطة، التوازن، تداخل البيانات توزع أجزاء من نفس البيانات على: (Data parallelism) توازي البيانات: (Types of Parallelism) الأخطاء. أنواع التوازي توزع الخيوط عبر النوى، بحيث يقوم كل خيط بتنفيذ (Task parallelism) عدة أنوية مع تنفيذ نفس العملية عليها. توازي المهام (User Threads and Kernel Threads) خيوط المستخدم (User Threads) عملية فريدة. 4. خيوط المستخدم وخيوط النواة وتتميز بسرعة الإنشاء (User-level threads library) تتم إدارتها بواسطة مكتبات خيوط على مستوى المستخدم (Threads): تدعمها وتديرها مباشرة نواة نظام: (Kernel Threads) ولكن قد تؤدي حجب خيط واحد إلى حجب العملية كلها. خيوط النواة: (Many-to-One) نموذج (Multithreading Models) التشغيل (مثل أنظمة ويندوز، لينكس، وماك). 5. نماذج تعدد الخيوط ربط كل خيط: (One-to-One) ربط العديد من خيوط المستخدم بخيط نواة واحد (خيط واحد محجوب يحجب الجميع). نموذج مستخدم بخيط نواة مقابل، مما يوفر توازياً أكبر ولكن بحد أقصى لعدد الخيوط لتجنب عبء النظام (مثل ويندوز ولينكس). السماح بتعدد خيوط المستخدم بالارتباط بعدد مناسب أو أقل من خيوط النواة. 6. مكتبات الخيوط: (Many-to-Many) نموذج لإنشاء وإدارة (API) مكتبات الخيوط: توفر للمبرمج واجهة برمجة تطبيقات (Thread Libraries & Issues) وقضايا متعلقة بها (Java) وخيوط جافا، (Windows threads) خيوط ويندوز، (UNIX) المستخدم في أنظمة Pthreads الخيوط، وأبرزها معيار عند استخدام الخيوط. `fork()` و `exec()` دلالات الأوامر: (Threading Issues) أبرز القضايا والمشكلات. (threads) سواء بشكل (Thread cancellation) المتزامنة وغير المتزامنة. إلغاء الخيط المستهدف (Signal Handling) معالجة الإشارات تُنفذ: (Windows Threads) خيوط ويندوز 1. (Thread-local storage - TLS) فوري أو مؤجل. التخزين الخاص بالخيط على مستوى النواة. يحتوي كل (One-to-One) وتعتمد نموذج التعيين (Windows API) واجهة برمجة التطبيقات الأساسية مجموعة مسجلات، مكدسات منفصلة لبيئة المستخدم والنواة، (Thread ID) مستوى النواة. يحتوي كل خيط على معرف خاص لنواة النظام، و) `KTHREAD`، (للكثف) `ETHREAD`: تستخدم هياكل بيانات رئيسية في النواة مثل (TLS). خاصة لا تفرق لينكس بشكل كبير بين العمليات والخيوط؛ بل تشير إليها عادة: (Linux Threads) لبيئة الخيط). خيوط لينكس) `TEB` (نواة النظام، و) والذي يسمح بتحديد مستوى مشاركة `clone()` مهام). يتم إنشاء الخيوط أو المهام باستخدام استدعاء النظام Tasks باسم خيوط ويندوز 1. `struct task_struct` مساحة الذاكرة والبيانات بين العملية الأب والمهام الفرعية عبر مؤشرات (مثل على (One-to-One) وتعتمد نموذج التعيين (Windows API) تُنفذ واجهة برمجة التطبيقات الأساسية: (Windows Threads) مجموعة مسجلات، مكدسات منفصلة لبيئة المستخدم والنواة، (Thread ID) مستوى النواة. يحتوي كل خيط على معرف خاص `KTHREAD`، (للكثف) `ETHREAD`: تستخدم هياكل بيانات رئيسية في النواة مثل (TLS). ومنطقة تخزين بيانات خاصة لا تفرق لينكس بشكل كبير بين العمليات والخيوط: (Linux Threads) لبيئة الخيط). خيوط لينكس) `TEB` (نواة النظام، و) والذي يسمح بتحديد `clone()` مهام). يتم إنشاء الخيوط أو المهام باستخدام استدعاء النظام Tasks بل تشير إليها عادة باسم

struct task_struct). مستوى مشاركة مساحة الذاكرة والبيانات بين العملية الأب والمهام الفرعية عبر مؤشرات (مثل