

## « OS 1 »

Operating system : يوفر نظام التشغيل البيئة التي يتم من خلالها تنفيذ البرامج .. سوف نستعرض خدمات النظام مكونات وترابطه

## Operating-system services

① User interface واجهة المستخدم : تحتوي على أنظمة التشغيل على User interface (UI) يمكن أن تتخذ هذه الواجهة عدة أشكال واحدة من هذه الأشكال هي Command-Line interface (CLI) والتي تستخدم أوامر نصية وطريقة الإدخال (مثال : لوحة مفاتيح كتابة الأوامر بتتسلف محدد مع خيارات محددة) Shell : هي برنامج يعرف خدمات نظام التشغيل المستخدم أو برنامج آخر ، يستخدم shells إما (CLI) أو (GUI) graphical user interface (GUI) : هو الأكثر استخدام وشيوع تعتمد على الرسومات وتعامل بشكل مباشر مع I/O

② Program execution تنفيذ البرامج : يجب أن يكون النظام قادراً على تحميل البرنامج في الذاكرة وتشغيل هذا البرنامج يجب أن يكون قادراً على إنهاء تنفيذه أو عدم تنفيذه (بشير الخطأ)

③ I/O operations عمليات الإدخال والإخراج : لا يستطيع المستخدمون عادة التحكم بأجهزة الإدخال والإخراج مباشرة لذلك يجب أن يوفر نظام التشغيل وسيلة للقيام بذلك .



مانر بلايين

④ file-system manipulation : إن نظام الملف ذو أهمية خاصة (obvi-ously) تتطلب البرامج إلى قراءة وكتابة الملفات والادارة كما يحتاجون أيضاً إلى انشائها وحذفها بالرسم والبحث عن ملف معين ومعلومات ، تخزين بعض أنظمة التشغيل ادارة الازونات للسماح بالوصول إلى الملفات أو المحلل أو رفعه بناداً على ملكية الملفات .

⑤ communications الاتصالات : هناك بعض العمليات التي يحتاج فيها إلى تبادل معلومات مع عملية أخرى ، وقد يحدث هذا الاتصال بين العمليات التي يتم تنفيذها على نفس الجهاز أو بين العمليات التي يتم تنفيذها على أنظمة كومبيوتر مختلفة مرتبطة مع بعضها بواسطة شبكة ، يمكن تنفيذ الاتصالات عبر shared memory حيث تقوم عمليات أو أكثر بالقراءة والكتابة إلى قسم مشترك من الذاكرة . أو عن طريق (message passing) حيث يتم نقل حزم المعلومات في تنسيقات محددة مسبقاً بين العمليات بواسطة

تكنيكين

⑥ Error detection كشف الأخطاء : يحتاج دة إلى الكشف والتدقيق عن الأخطاء باستمرار . أخطاء قد تحدث في CPU وأجهزة الذاكرة (مثل انقطاع التيار الكهربائي) أو في أجهزة I/O (مثل الاتصال بالشبكة) يجب أن يتخذ دة الإجراء المناسب في بعض الأحيان ليس لديها خيار آخر سوى إيقاف النظام



هناك وظائف أخرى ليس بمساعدة المستخدم بل لضمان التشغيل الفعال للنظام نفسه :-

① Resource allocation تفصيل الموارد :-  
عندما يكون هناك عدة مستخدمين أو وظائف متعددة تعمل في نفس الوقت يجب تفصيل الموارد لكل منهم

② Accounting المحاسبة :- عندما نريد تتبع مقدار وأنواع الموارد المستخدمة فنستخدم حفظ السجلات للمحاسبة ( بحيث يمكن إحصاء فواتير المستخدمين ) أو ما جرد تجميع للموارد المستخدمة

③ Protection الحماية :- هي آلية للتحكم في وصول العمليات أو المستخدمين إلى الموارد المحددة بواسطة نظام المحاسبية .

④ security الأمان :- هو الرفاع عن النظام من الهجمات الخارجية والداخلية . مثل الفيروسات والديدان وهجمات رفض الخدمة وسرقة الهوية والخرقة .



System Calls : هو وسيلة لطلب الخدمات من كس  
 متى يتم تنفيذها ، كما أنها توفر واجهتين المهم ونظام  
 التشغيل ، تتوفر هذه الطلبات بشكل عام كإجراءات مكتوبة  
 بلغة C و C++ على الرغم من أن بعض المهم ذات المستوى  
 المنخفض قد يتعين كتابتها باستخدام ~~لغة التجميع~~  
 إجراءات لغة التجميع (assembly language)

Application programming interface (API) . هو وصف المتاح  
 (API) interface . البرمجة حسب وظائفها  
 ومدخلاتها ومخرجاتها ويحدد الهدف الرئيسي منها  
 في توفير قائمة من الوظائف المستقلة تماماً عن  
 الآلية التي نفذت بها ، لتتيح للآخرين التوافق معها  
 من خلال أي آلية أخرى . معظم تفاعيل واجهة كس مخصصة  
 من المبرمج والمستخدم بواسطة API .  
 System-call interface : يعد بمثابة رابط  
 لـ system calls التي يوفرها كس وتعرف واجهة استدعاء  
 النظام استدعاءات الوظائف في ملف API ويستدعي  
 استدعاءات النظام الضرورية داخل نظام التشغيل .

نقلت لحرق عامة تستخدم لتعريف البراميترس الى نظام التشغيل

① تمر Parameters في registers و هو أبسط نهج

② في بعض الأحيان تكون هناك Parameters أكثر من

Register فيتم تخزين البراميترس بشكل عام في block  
 أو جدول في الذاكرة و يتم تمرير عنوان block  
 كبراميترس في الرد



③ يمكن أيضاً وضع البراميترس او stack is pushed بواسطة البرنامج و popped من ال stack بواسطة نظام التشغيل OS.

٢/ تفقد بعض أنظمة التشغيل طريقة

Prefer the block or stack method parameters?  
(مع) لأن هذه الأساليب لا توفر من عدد أو طول البراميترس التي يتم تمريرها

## Types of system calls :-

- ① Process control
- ② File manipulation
- ③ device manipulation
- ④ information maintenance
- ⑤ communications
- ⑥ Protection

حماية المعلومات

حماية

system utilities

System Programs : - توفر بيئة ملائمة لتشغيل

البرامج وتنفيذها بفعالية مجرد واجهات مستخدم

ال system calls والبعض الآخر أكثر تعقيداً بكثير

ويمكن تقسيمها إلى



⑦ File management : تقوم هذه البرامج بإنشاء  
 حذف ، نسخ ، إعادة تسمية ، طباعة ، تغريغ ، سرد  
 والتعامل بشكل عام مع directories and manipulate files

⑧ status information : يعرف البرامج ~~معلومات~~  
 تطلب معلومات من النظام تاريخ الوقت أو مقدار الذاكرة  
 المتوفرة أو عدد المستخدمين اليه في الأخير الأكثر تعقيداً  
 حيث يوفر أدوات مفيدة وسهلة ومعلومات تمهيدية الاخطاء  
 تدعم بعض الأنظمة ملف (registry) والتي تستخدم  
 لتخزين واسترجاع معلومات التكوين .

③ File modification : تعديل الملف : ويشمل تحرير النصوص  
 لإنشاء وتعديل محتوى الملفات

④ Programming - Language support : كوميلا ، اسمبلي  
 والمترجمون الفوريون للغات البرمجية (C, C++, Java)  
 تقوم غالباً مع OS أو متاعاً كتنزيل منفصل .

⑤ Program loading and execution : تحميل البرنامج وتنفيذه  
 بمجرد تحميل البرنامج يجب تحميله في الذاكرة ليتم تنفيذه .  
 يجد يوفر هذا النظام لودر وحلقة ولودر قابلة للنقل  
 ومحرران ربط ولودر مركبة هناك حاجة أيضاً إلى  
 أنظمة تمهيد للغة عالية المستوى وحلقة الماكينة .



⑥ Communications : توفر هذه البرامج آلية لإنشاء اتصالات افتراضية بين العمليات والمستخدمين وإنفاذ الكمبيوتر.

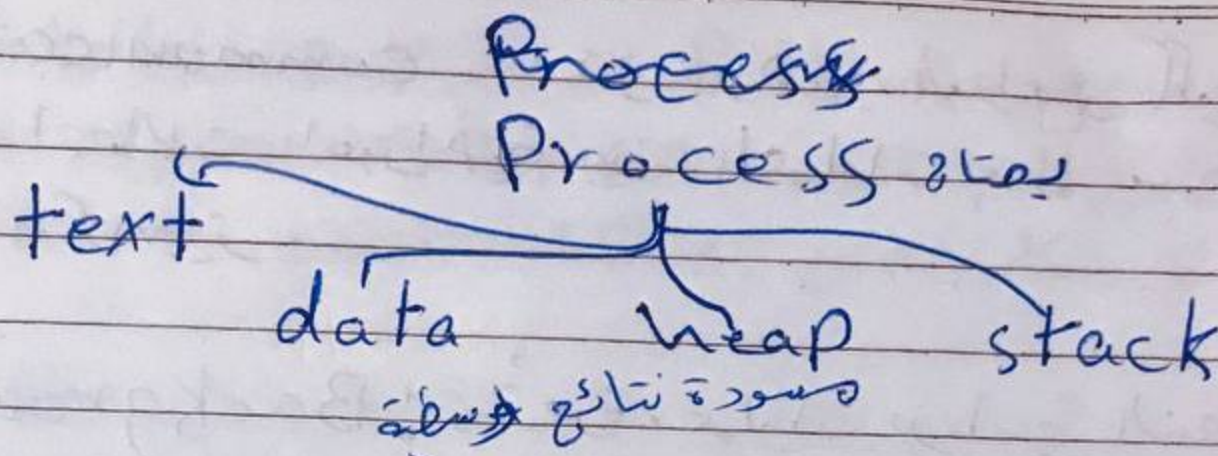
⑦ Background : تُعرف عمليات برنامج النظام services الجارية باستمرار باسم (services) أو (subsystems) أو (daemons)

## « 2 ك 0 »

Process : عبارة عن برنامج قيد التنفيذ هي أكثر من Code للبرنامج والذي يُعرف أيضاً بـ (text section) يتضمن أيضاً النشاط الحالي وتمثل قيمة لـ (Program Counter) ومحتويات سجلات البروسس. تتضمن البروسس أيضاً عملية الـ (stack) والتي تحتوي على بيانات مؤقتة مثل (Function Parameters, return address) وتحتوي أيضاً على (data section) أي تحتوي على متغيرات عالمية. تحتوي البروسس أيضاً على (heap) وهي الذاكرة التي يتم تخصيصها ديناميكياً أثناء وقت تشغيل البروسس.

executable file : هو ملف يحتوي على قائمة من التعليمات instructions محفوظة على disk وعبر برنامج يديرها إشارات النظام للتنفيذ ومجموعة من ~~الموارد~~ الموارد.





Process state : عندما يتم تنفيذ البرنامج  
تغير حالة (state) يتم تحديد حالة البرنامج، جزئياً بواسطة  
التيار الذي لك العملية.  
هناك حالتان للعملية :

- 1- new : انشاء او بداية ال Process
- 2- Running : يتم تنفيذ الاوامر (Instruction)
- 3- waiting : ينتظر البرنامج حدوث بعض event
- 4- Ready : ~~البرنامج ينتظر~~ يتم اختيار البرنامج
- 5- Terminated : انتهى البرنامج من التنفيذ

Process control Block (PCB) : يتم  
تمثيل كل برنامج في OS بواسطة (PCB) الذي يدير ايها  
ب (task control Block). يحتوي على العديد من المعلومات المرتبطة  
في Process من بينها ذلك :

- 1- Process state : قد تكون الحالة new او  
halted & waiting



2- Process counter :- يشير الى الاعازات التالية التي سيتم تنفيذها لهذا البروسس .

3- CPU Register :- تختلف الريجستر من حيث العدد والنوع اعتماداً على بنية الكومبيوتر وهي تشمل (stack index) general-purpose register (counter) و خافعة الرأى معلومات عن condition-code.

4- CPU-scheduling information :- تتضمن هذه المعلومات أولوية البروسس ومؤشرات لجدولة قوائم الانتظار .

5- Memory-managment information :- تتضمن قيم الحسابات الأساسية والحدود و جداول الصفحات او جداول المقطع اعتماداً على نظام الذاكرة المستخدم بواسطة OS .

6- Accounting information :- تتضمن هذه المعلومات مقدار CPU والوقت الفعلي المستخدم ، والقيود الزمنية وارقام الحسابات وارقام الوظائف والعمليات .

7- I/o status information :- تتضمن معلومات I/O



**Threads :** هو عبارة عن مجموعة من العمليات التي تشكّل مساراً لتنفيذ العملية تجعل ال Thread البرنامج وكأنّه يتعدّد أكثر من مهمة بشكل متزامن . هذه الميزة مفيدة بشكل خاص في أنظمة multicore حيث يمكن تشغيل threads متعددة بشكل متوازٍ .

**Process scheduling :** الهدف من البرمجة المحدودة هو تشغيل بعض العمليات في جميع الأوقات ، لزيادة استخدام وحدة المعالجة المركزية ، الهدف من مشاركة الوقت هو تبديل CPU بين المهام البرمجية بشكل متكرر بحيث يمكن للمستخدم التفاعل مع كل البرنامج . وإيضا يختار البروسسور البرنامج (ربما يكون مجموعة من البروسسور المتناهي) لتنفيذ البرنامج على CPU . بالنسبة لنظام المعالج الواحد لن يكون هناك أكثر من عملية واحدة قيد التشغيل

**scheduling Queues :** عندما تدخل العمليات إلى النظام يتم وضعها في ملف (job queue) والتي تتكون من جميع العمليات في النظام . يتم الاحتفاظ بالعمليات الموجودة في الذاكرة الرئيسية والمجهزة للتنفيذ في قائمة تسمى (ready queue) يتم تخزين قائمة الانتظار هذه بشكل عام كقائمة مرتبطة . تسمى قائمة العمليات التي تنتظر جهاز I/O (device queue)



بمجرد تفرغ وحدة المعالجة المركزية للعمليات وتنفيذها  
يمكن ان يذهب امر الاوامر :-

① ان تمر عملية طلب I/O ثم يتم وضعها في قائمة انتظار I/O

② يمكن ان تعلق child process وتنتظر انتهاء ال child

③ يمكن ازالة البروسس بقوة من CPU ، نتيجة للقائمة  
واعادتها الى قائمة الانتظار  
سكحولرس

- schedulers المنظرون :- هو نظام software يسمح  
للاؤسسة بحرولة المهام وتوزيعها ويتم ذلك من قبل  
المجدول (schedulers)

Long-Term scheduler : يعرف ايضاً باسم  
(job scheduler) ينظم برنامج الجدولة على المدى الطويل  
في هذا يتم اعداد البرامج في قائمة الانتظار ووفقاً للمتطلبات  
يتم اختيار افضل وثيفة واحدة . يُحفظ في ال Hard .

short-Term : ينظم جدولة على المدى القصير  
استعمالاً أكثر ويحفظ في ال Ram و يُنفذ ب CPU  
وتنفذه اسرع



العملية

Context switch : هو تخزين الحالة العملية الحالية  
 لـ Process في PCB ومن ثم تحميل الحالة التخزينية  
 السابقة لـ Process (عملية تخزين وتحميل أثناء التنفيذ)

degree of multi programming : تعرف درجة البرمجة  
 المتعددة المر الاقصى لعدد العمليات التي يمكن لنظام  
 معالج واحد استيعابها بكفاءة

I/O bound Process : هو يقف وقته في عمليات  
 الإدخال والإخراج أكثر من العمليات

CPU bound Process : يولد طلبات I/O بشكل متكرر  
 غير متكرر باستخراجه المزيد من وقته في العمليات

medium term scheduler : هو مستوى أخا في  
 متوسط من الجدولة ، في بعض الأحيان قد يكون من  
 المفيد إزالة عملية من الذاكرة وبالتالي تقليل درجة البرمجة  
 المتعددة ، يمكن إعادة تقديم العملية في الذاكرة ومتابعة تنفيذها  
 حسب توقعات وهذا ما يسمى بـ (swapping)  
 حيث يتم تخزين العملية

~~status save~~