

# نظم التشغيل

المحاضرة رقم (4)

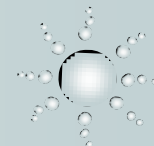
## إدارة العمليات

## Processes Management

د. أشرف الزبير محمد

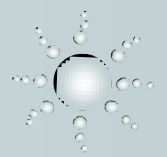
كلية دراسات الحاسوب

جامعة افريقيا العالمية



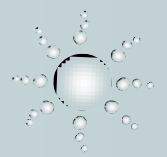
\* إدارة العمليات هو جزء هام من نظام التشغيل ويعتني بكل ما يتعلق بالعمليات من:

- العمليات (processes).
- الخيوط أو العمليات الخفيفة (Threads).
- جدولة المعالج (CPU scheduling).
- تزامن العمليات (Process synchronization).
- الاختناق بين العمليات (Deadlocks).

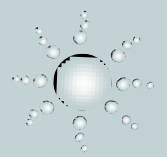


## \* مفهوم العملية (Process Concept)

البرنامج يكون في شكل ملف عندما يكون مخزن بالقرص الصلب (أو أي وسيط تخزين ثانوي) وعندما ننقر عليه نقرأ مزدوجاً فإننا نطلب من نظام التشغيل تنفيذه، فيقوم نظام التشغيل بتحميله من القرص الصلب (أو وسيط التخزين الموجود به مثل الفلاش أو الأسطوانة) إلى الذاكرة الرئيسية (الرام) لبدأ التنفيذ، هنا يتغير اسم البرنامج من ملف إلى عملية.

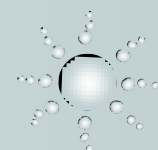


- العملية هي برنامج شغال (تحت التنفيذ)، أحيانا نطلق عليها عمل (job) أو مهمة (task).
- البرنامج هو سلسلة من الأوامر تعطى للحاسب للقيام بعمل ما. ينفذ البرنامج داخل المعالج تسلسليا، أمر تلو الآخر.

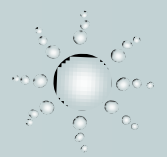


## \* خطوات تنفيذ البرنامج:

1. تحميل البرنامج في الذاكرة الرئيسية.
2. يتم وضع عنوان بداية البرنامج (عنوان أول أمر بالبرنامج) في مسجل داخل المعالج يسمى عداد البرنامج (PC program counter).
3. يقوم المعالج بإحضار أول أمر بالبرنامج من الذاكرة الرئيسية ويتم تخزينه في مسجل داخل المعالج.
4. زيادة عداد البرامج ليشير إلى الأمر الذي يليه.
5. فهم وتنفيذ الأمر الذي أحضرناه.

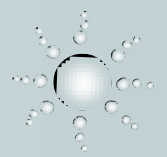


6. إحضار الأمر الذي يليه (الذي يشير له عداد البرامج)
7. زيادة عداد الأوامر ليشير إلى الأمر الذي يليه.
8. فهم وتنفيذ الامر الذي بالمعالج.
9. انتقل إلى الخطوة 6، وهكذا نكرر هذه الخطوات إلى أن ينتهي تنفيذ البرنامج.



## \* حالات العملية (process states)

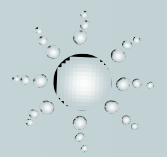
تحميل البرامج في الذاكرة يجعل هذه البرامج جاهزة للتنفيذ (ready)، عند بداية تنفيذ البرنامج داخل المعالج يصبح شغال (running)، قد يستمر المعالج في تنفيذ البرنامج حتى يكتمل، وقد يوقف المعالج البرنامج الشغال (مؤقتا) لسبب ما، فيصبح البرنامج في هذه الحالة محجوز (blocked)، وقد يشتغل برنامج آخر أكثر أهمية (مثلا).



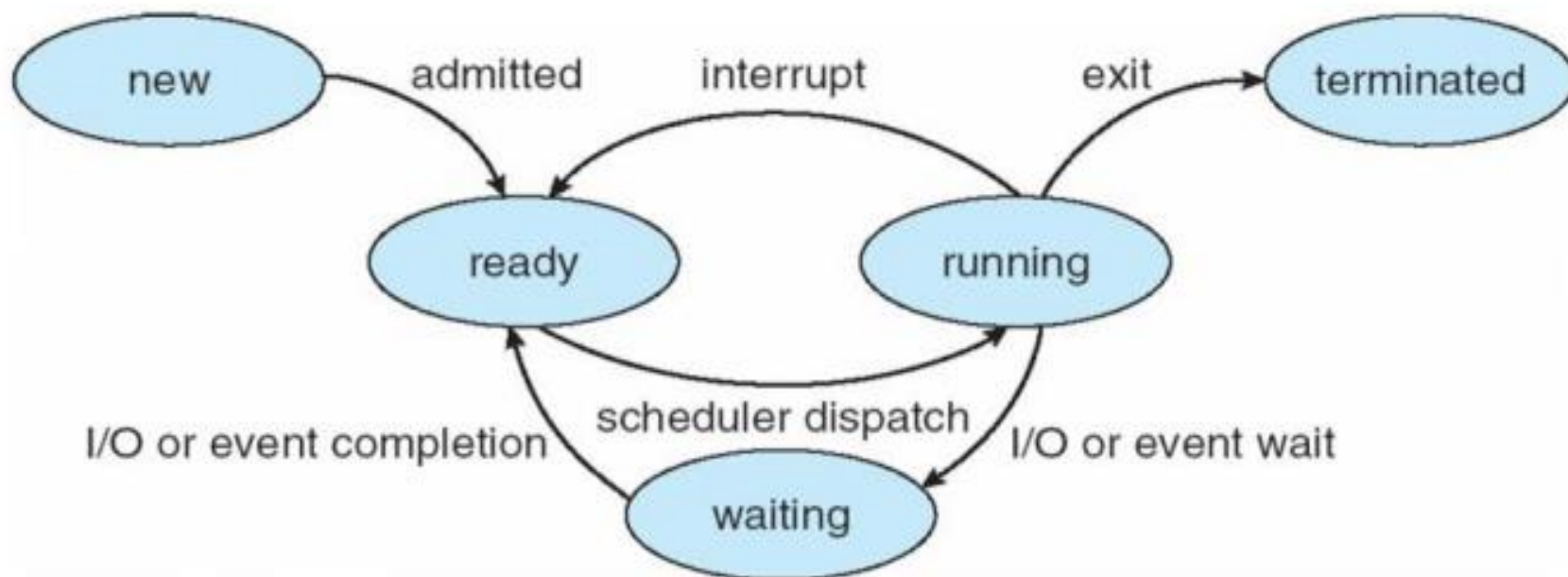
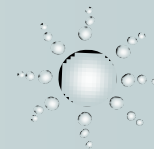
\* إذن تحميل البرنامج بالذاكرة يسمى عملية، هذه العملية (أو البرنامج) يتغير وضعها من حال إلى حال، كما موضح أدناه:

- جديد (new): العملية تم إنشاءها وجاهزة للتحميل.
- حالة الجاهزية (ready state): العملية تم تحميلها في الذاكرة وأصبحت جاهزة للتنفيذ.
- حالة التنفيذ (running state): العملية بدأت التنفيذ داخل المعالج (يتابع مسجل عداد البرامج تسلسل تنفيذ أوامر العملية).

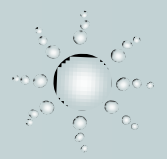




- حالة الحجز أو الانتظار (blocked state or waiting): عندما يوقف المعالج عملية، تصبح هذه العملية محجوزة. يتم توقيف العملية لأسباب عدة مثل الحاجة لتشغيل عملية أخرى أكثر أهمية، أو أن العملية تنتظر حدث (event) معين لم يتم بعد، أو أن الزمن الذي خصص للعملية قد اكتمل (المشاركة الزمنية).
- الانتهاء (terminated): هنا تكون العملية قد انتهى عملها، فتقوم بإخلاء طرفها (تحرير الموارد التي كانت تستخدمها، وإخلاء الذاكرة التي كانت تحتجزها) قبل الخروج.



شكل : حالات العملية .

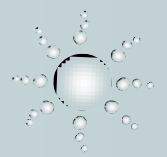


## \* إنشاء العملية

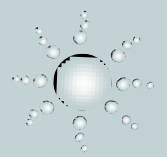
عند إنشاء العملية تعرف وتدار برقم غير متكرر يسمى رقم تعريف العملية (Process Identification Number (PID)).

\* هنالك أسباب مختلفة لإنشاء العملية مثل:

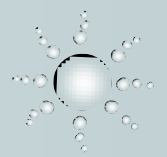
- تهيئة النظام: عند إقلاع نظام التشغيل تنشأ العديد من العمليات، منها ما يعمل في الخفاء (background)، ومنها ما يعمل ويتخاطب مع المستخدم.



- عملية تستدعي النظام لإنشاء عملية أخرى: أحيانا تقوم عملية منفذة بتشغيل (إنشاء) عملية أخرى تساعد في عملها، تعتبر العملية الأولى أب (parent) للعملية الثانية والتي تعتبر ابن (child)، العملية الابن يمكنها إنشاء عمليات أبناء لها مما قد يكون شجرة من العمليات. قد تعمل هذه العمليات معا في وقت واحد أو تنتظر بعضها البعض.
- طلب المستخدم إنشاء عملية جديدة: عند ما ينقر المستخدم نقرا مزدوجا على أيقونة برنامج فهذا طلب من المستخدم لإنشاء عملية جديدة.



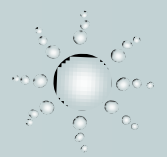
- المهام المحزمة (batch): قد يضع المستخدم حزمة من العمليات ويطلب من نظام التشغيل تنفيذها مرة واحدة، فيقوم النظام بتنفيذ العملية الأولى في الحزمة، ثم متى ما أتاحت له موارد يقوم بتنفيذ العملية الثانية وهكذا إلى أن ينفذ كل العمليات الموجودة بالحزمة.



## \* إنهاء العملية

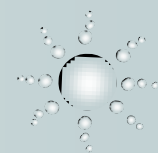
- بعد تنفيذ العملية لآخر أمر فيها ستطلب من نظام التشغيل أن يقوم بحذفها وذلك بإستدعاء نداء النظام `exit` مثلاً.

- مخرجات العملية المحذوفة ترسل للعملية الأب عبر استدعاء النظام `wait`، بينما يقوم نظام التشغيل بتحرير كل موارد العملية .



\* قد يقوم الأب بإنهاء العملية الابن (abort) منها:

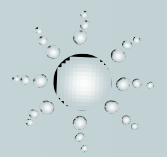
- زاد الابن في عدد الموارد المخصصة له.
- لم يعد الأب يحتاج لما يقوم به الابن (المهمة المنفذة لم نعد بحاجة لها).
- إذا أنهى الأب عمله exiting.
- بعض نظم التشغيل لا تسمح للأبن بمواصلة التنفيذ إذا أنهى الأب عمله.
- كل الأبناء سينتهون بانتهاء الأب cascading termination.



## \* أسباب إنتهاء العملية:

- انتهاء طبيعي (إكتمل عملها).
- الانتهاء بسبب حدوث خطأ.
- تم إنهاءها بعملية أخرى .

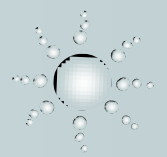




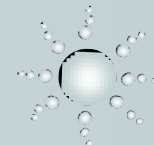
## \* معلومات العملية (Process Control Blocks (PCB)

لكل عملية بنية بيانات (data structure) تسمى (PCB) تخزن فيها المعلومات الأساسية للعملية، شكل رقم (3-4)، تجمع البنيات الأساسية لكل العمليات في جدول يسمى جدول العمليات (process table)، حيث يكون هنالك خانة لكل بنية عملية بالنظام. تحتوي بنية العملية على معلومات عن العملية مثل:

- رقم تعريف العملية (process identification).
- حالة العملية (process state).

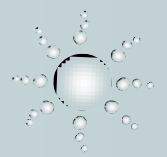


- محتوى عداد البرامج (Program counter).
- مسجلات المعالج CPU registers.
- معلومات جدولة المعالج CPU scheduling information.
- معلومات إدارة الذاكرة Memory-management information.
- معلومات الحسابات Accounting information.
- معلومات حالات الدخل والخرج I/O status information.
- مقدار ما نفذ من العملية.
- مكان الذاكرة المستخدم من قبل العملية.
- الموارد التي تستخدمها العملية مثل الملفات المفتوحة بواسطة العملية.
- أولية العملية.



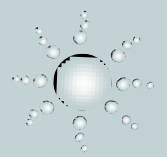
مؤشر	حالة العملية
	رقم العملية
	عداد البرامج
	محتوى المسجلات
	حدود الذاكرة
	الملفات المفتوحة
	.....

شكل : بنية معلومات العملية (PCB).



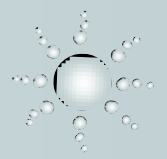
## \* التحول السياقي (Context switch)

- تستخدم معلومات العمليات عندما تتحول العملية من حالة التنفيذ إلى حالة الجاهزية أو الحجز. عندها يقوم المعالج بتوقيف عملية وتنفيذ عملية أخرى، حيث يتم حفظ معلومات العملية التي تم توقيفها (مثلا العملية P0) في PCB0، ثم يتم تحميل معلومات العملية المراد تنفيذها (مثلا P1) من PCB1. إذا أراد نظام التشغيل تنفيذ P0 مرة أخرى فسيقوم بتخزين معلومات P1 في PCB1 ثم تحميل معلومات P0 من PCB0 حيث يستطيع مواصلة التنفيذ من آخر نقطة وقفت فيها العملية P0.
- الزمن المستغرق في الانتقال بين عمليتين يكون مهدور وغير مستفاد منه، حيث لا يقوم النظام بعمل في هذه الفترة.



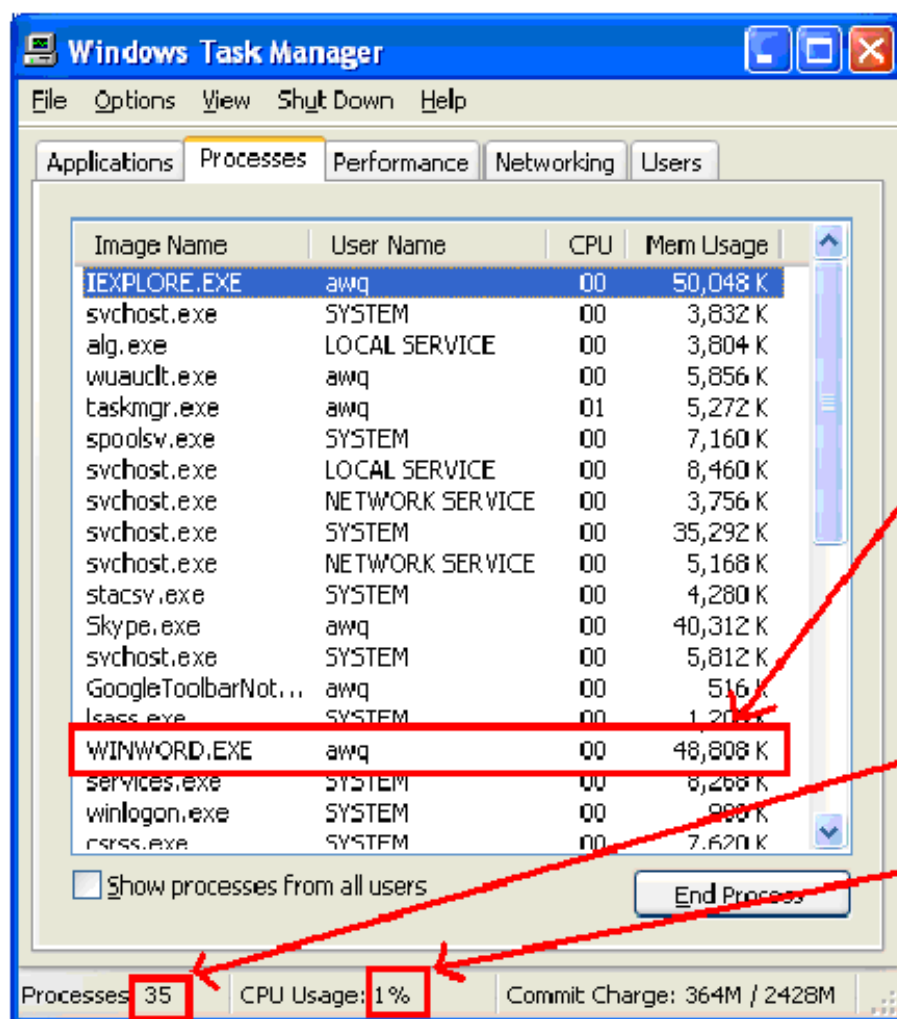
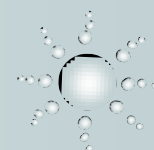
## \* العمليات في ويندوز

- عندما ننفذ برنامجا على ويندوز، لابد لنظام التشغيل من معرفة كيف يدير هذه البرامج للتأكد من أن كل برنامج أخذ حصته من الوقت في المعالج وفي الوصول إلى الذاكرة وأجهزة الدخل والخرج.
- ولتحقيق ذلك يتعامل نظام التشغيل مع كل برنامج كعملية.
- فإذا قمنا بتشغيل برنامج فسينشئ عملية لهذا البرنامج، وإذا نفذنا نسخة من نفس البرنامج فسيقوم نظام التشغيل بإنشاء عملية أخرى لهذه النسخة، بحيث تكون هنالك عملية لكل نسخة شغالة من البرنامج.



- وبالتالي كل البرامج التي تعمل في نظامك هي عبارة عن عمليات يدير ويتابع عملها نظام التشغيل.

- لمعرفة العمليات التي تنفذ بجهازك حالياً قم بالضغط على `Ctrl+Alt+Del` فتظهر نافذة، انقر على تبويب `Processes` فترى كل العمليات التي تعمل الآن في جهازك بما فيها عمليات نظام التشغيل ومضادات الفيروسات والبرامج الخدمية ، وترى حجم الذاكرة الذي تستخدمه كل عملية، الشكل (3-3).

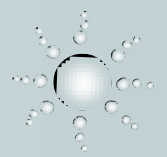


مثلا أنا الآن أفتح وورد

عدد العمليات  
الموجودة بالنظام

نسبة استخدام المعالج

شكل رقم (3-3): مشاهدة العمليات في ويندوز.

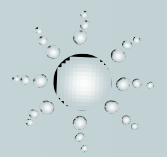


- من الشكل (3-3) نجد أن كل عملية لديها مالك (user name) هو الذي شغلها.

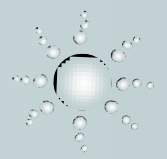
- أيضا يظهر تحت CPU، زمن المعالج الحالي المستخدم لكل عملية، كذلك المساحة المستخدمة من الذاكرة والتي تخزن فيها العملية شفراتها وبياناتها (Mem Usage).

- مثلا برنامج وورد (WINWORD.EXE) هو عملية قام بتشغيلها المستخدم awq، وتستخدم ذاكرة حجمها 48,808k.

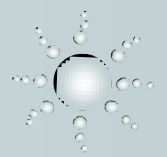




- يمكنك النقر على أي عملية وإيقافها من مدير المهام بالنقر على العملية ثم النقر على الزر Process End.
- مع التنبيه إلى أن كل مستخدم يستطيع إيقاف عملياته، إذا كان مستخدم عادي، ولا يستطيع إيقاف عمليات المستخدمين الآخرين إلا إذا كان مشرف Administrator.



- بالرغم من أن ويندوز تبدو لك أنها تشغل أكثر من برنامج (عملية) في نفس الوقت، إلا أن هذا غير صحيح للأجهزة ذات المعالج الواحد.
- فالحقيقة أن عملية واحدة فقط تعمل في الوقت الواحد، ثم تخرج عندما تكون عاطلة (idle)، وتنفذ عملية غيرها.
- أما في الحاسبات التي تمتلك أكثر من معالج فيمكن تشغيل أكثر من عملية، بحيث تشتغل كل عملية في معالج وفي نفس الوقت.



## \* الاتصال بين العمليات

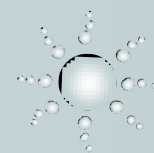
- قد تكون العمليات الموجودة في النظام مستقلة أو متعاونة (or cooperating independent).

- العمليات المتعاونة تؤثر وتتأثر بما حولها من عمليات، أما العمليات المستقلة فلا.

العمليات المتعاونة تحتاج اتصال فيما بينها (communication) interprocess، حيث يوجد نوعين من طرق الاتصال، هي:

1/ وجود ذاكرة مشتركة Shared memory

2/ عن طريق تبادل الرسائل Message passing



شكراً جزيلاً



أي سؤال