# الباب السابع

# ٧

# المفاهيم المتقدمة فى أدارة الذاكرة الاساسية

تعرضنا فى الباب السادس الى التصفح كأسلوب للتعامل مع الذاكرة الأساسية ضمن إطار الحيز غير المتصل وذكرنا شرطاً أساسياً أن حيز الصفحة يجب أن يكون مساوياً تماماً لحيز الإطار ، وعلى ضوء هذا الاعتبار يتولى برنامج مدير الذاكرة القيام بالأعمال التالية:

أ ــ تحديد عدد صفحات البرنامج.

ب ــ يحدد عدد كاف من أُطر فى الذاكرة أساسية تناسب عدد الصفحات ولا يشترط التلامس الفيزيائى فيما بينها.

ج ــ يتولى أدارة وتحميل الصفحات إلى الأطر المخصصة لها.

وبذلك يتم شحن البرنامج وتجدر الإشارة إلى أمر حيوى بالنسبة للقارئ أن عملية تقسيم البرنامج إلى صفحات تتم وهو لازال مستقراً فى ذاكرة القلب المصورة آخذين فى الإعتبار أنه من الأفضل والأسهل والأيسر لنظم التشغيل أن يكون حيز القطاع Sector على وسائط التخزين الثانوية مساوياً تماماً لحيز الصفحة مما يحقق سيطرة مزدوجة من الكيان البرمجى لنظام التشغيل على عدد الأحرف المنقولة للصفحة.

ورغم أن أسلوب التصفح يحقق كفاءة جيدة فى أدارة الذاكرة الأساسية إلا أنه يأتى أيضا بمشاكل جديدة ، أبرزها:

أ ــ ضرورة أن يتعرف مدير الذاكرة على كل الصفحات وحالة كل صفحة وهذا يعنى زيادة تعقيد برامج التشغيل مما يسبب تكلفة اضافية على النظام.

ب ــ أحتمال ظهور مشكلة التفتيت الداخلى Internal Fragmentation فى الصفحة الأخيرة من البرنامج.

ج ــ فتح جداول كثيرة تستهلك جزءاً لا يستهان به من الذاكرة الأساسية.

د ــ كلما طلب برنامج التطبيقات أستدعاء بايت واحدة أو سطر إيعاز من الصفحة يلزم أولاً لنظام التشغيل التعرف على الصفحة من خلال الجداول ثم حساب الإزاحة المطلوبة لتحديد عنوان الايعاز داخل الصفحة ، مما يستغرق وقتاً من وقت التشغيل الفعلى.

ويوضح المثال التالى مضمون ومفهوم هذه العيوب:

**مثـــال (١):**

يشغل أحد برامج التطبيقات حيزاً قدرة ٤٥٠ بايت فما هى المشاكل الفنية التى تنجم عن هذا البرنامج تحت ظروف التصفح بفرض أن حيز الصفحة يساوى مائة بايت؟

**الحـــل:**

١ ــ حيث أن حيز الصفحة مائة بايت ، فأن عدد الصفحات المطلوبة يساوى ٤٥٠/١٠٠ = ٤,٥ صفحة.

ولأنه لايمكن أستخدام نصف صفحة ، بالتالى يلزم خمس صفحات ولا يشغل من الصفحة الخامسة إلا نصف حيزها مما يسبب مشكلة التفتيت الداخلى.

٢ ــ تتطلب متابعة أداء هذا البرنامج ثلاثة جداول هى:

أ ــ جدول المهام JOB TABLE (J.T) .

ب ــ جدول تطابق الصفحات والأطر PAGE MAP TABLE (PMT) .

ج ــ جدول تطابق الذاكرة MEMORY MAP TABLE (MMT) .

ويسمى جدول حالة الأطر والموضحة فى الإطار التالى:

PMT

| رقم الصفحة للمهمة | رقم إطار الصفحة |
|---|---|
| صفر | ٨ |
| ١ | ١٩ |
| ٢ | ٣٧ |

J . T

| حيز المهمة | عنوان أول صفحة فى الذاكرة لهذا العمل |
|---|---|
| ٤٥٠ | ٣٠٩٦ |
| ٣١٠٠ | ٧٥٠ |
| ٤٠ | ٣١٥٠ |

MMT

| رقم إطار الصفحة | الحالة |
|---|---|
| ٨ | يعمل |
| ١٩ | عاطل |
| ٣٧ | فارغ |

والتعامل مع مثل هذا الكم من الجداول والإرتباطات يفرض على النظام تكلفة إضافية.

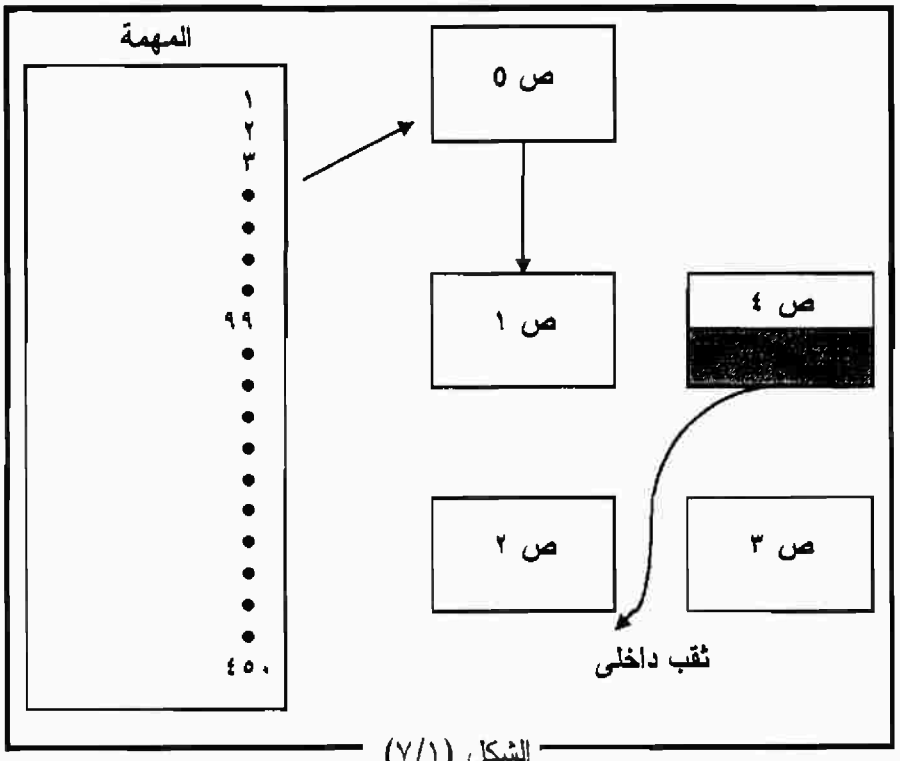
**مثال (٢):**

بفرض أن كل صفحة فى المثال السابق تحتوى على أكثر من سجل RECORD ويحتاج البرنامج إلى سجل بعينه فأنه يجرى حساب الإزاحة المطلوب ، على النحو الموضح فى الشكل (٧/١).

أ ــ يحسب عدد الصفحات ويتضح أنها خمس صفحات مع وجود ثقب في الصفحة رقم (٤).

ب ــ يحدد موقع السجل ويحدد الصفحة المحتوية على السجل ولتكن ص ٤.

ج ــ يحدد موقع السجل فى الصفحة الثانية بحساب عنوان أول بايت.




الشكل (٧/١)

ويطلق على الإزاحة OFFSET أو DISPLACEMENT.. مع العلم أن إزاحة السطر الأول أو السجل الأول أو البايت الأولى فى أى صفحة يساوى صفراً فان الإزاحة التالية تساوى واحد والإزاحة الثالثة ٢ وهكذا فأن أقصى إزاحة فى الصفحة

الأولى تساوى ٩٩ ، وعليه يقوم نظام التشغيل بإحضار رقم الصفحة مستخدما جداول النظام ثم يحسب الإزاحة ويصل الى عنوان الموقع للسجل المطلوب.

وهذه الحسابات عملية رياضية تستهلك وقتاً من المشغل مما يزيد من التكلفة الفعلية للنظام ، كل هذه المشاكل حاول المتخصصون فى نظم التشغيل علاجها عن طريق استحداث أسلوب جديد هو أسلوب الصفحات المطلوبة ونعنى بها الصفحات التى تحتوى على إيعازات يشتد عليها الطلب.

## أسلوب الصفحات المطلوبة DEMAND PAGE:

يستخدم هذا الأسلوب فكرة التبديل SWAPPING فى إدخال الصفحات المطلوبة إلى الذاكرة الأساسية وليس شحن كل صفحات البرنامج كما هو موضح فى الشكل (٧/٢) والذى يمثل شاشة تضم عدة خيارات أمام المستخدم وله أن يختار منها خياراً واحداً والذى يفهم منه أنه مادام الحاسب لا يمكنه سوى تنفيذ عمل واحد فأن من الأفضل إدخال الصفحات التى تشمل إيعازات الوظيفة المطلوبة فقط.

صيانة النظام

حدد وظيفة واحدة :

١ - إضافة سجل جديد .

٢ - حذف سجل .

٣ - تحديث سجل .

٤ - عودة للشاشة السابقة.

= أدخل الإختيار المطلوب ... .. ( )

إحدى شاشات نظام معلومات

الشكل (٧/٢)

وقد حقق هذا إمكانية تشغيل عدد أكبر من المهام فى أقل حيز من الذاكـــرة الأساسية مقارنة بنظام التصفح السابق مما أعطى لمستخدمى الحاسب وهما بأن سعة الذاكرة الأساسية لا نهائية ودفع العلماء نحو التفكير فى أسلوب الذاكرة التخيلية.

وكشأن كل نظم المعلومات فأن أسلوب "الصفحات المطلوبة" يرتكز أساســاً على وجود وسائط تخزين ثانوية سريعة جداً DASD حيث يتم تبادل الصفحات بينها وبين الذاكرة الأساسية.

ويستخدم أسلوب الصفحات المطلوبة نفس الجداول الثــلاث - J.T - PMT) (MMT السابق الإشارة إليها فى أسلوب التصفح مع إضافة ثلاث حقول جديدة هى:

أ ــ حقل يوضح فيه هل الصفحة مازالت فى إطارها بالذاكرة الأساســية ، فــإذا كانت الإجابة نعم.. فأين هذه الصفحة.. بمعنى أين موقعــها وبالتــالى يجــب تحديد موقعها.

ب ــ الحقل الثانى يوضح به هل حدثت تعديلات على محتويات الصفحــة ، فــإذا كانت الإجابة [نعم] فإنه يلزم إعادة كتابتها إلى الذاكرة الثانوية DASD أمــا إذا كانت الإجابة [لا] فإن الصفحة موجودة بالفعل على الأقراص المغناطيســية. وليس هناك مبرر لاعادة كتابتها بل يمكن أستخدامها كحيز متــاح مبــاشرة بالكتابة عليها صفحة جديدة.

ج ــ الحقل الثالث.. ويوضح هل تم طلب هذه الصفحة مجــدداً.. وذلــك لتحديــد الصفحات التى يشتد عليها الطلب فاذا كانت الإجابة [نعم] فإن نظام التشغيــل يفضل الإبقاء عليها فى الذاكرة وعدم إحلالها.

**مقاطعة الصفحة PAGE FAULT :**

سبق وأن أوضحنا فى باب سابق أنواع المقاطعات ، ونضيف إليها هنا نوعاً جديداً من المقاطعات يسمى "مقاطعة صفحة PAGE INTERRPUT" بمعنــى لــو أن برنامجاً يجرى تنفيذه طلب صفحة ولم تكن موجودة فى الذاكرة الأساسية فإنه يلجــأ

إلى عملية مقاطعة على وحدة التشغيل المركزية ومن ثم يتولى نمط خاص يسمى PAGE INTERRPUT HANDLER تحديد عنوان الصفحة المطلوبة ومادامت غير موجودة فإن النمط يحدد هل هناك إطار خال فى الذاكرة يطبع فيه الصفحة المطلوبة فإن كانت كل الأطر مشغولة فإن الروتين يحدد الإستراتيجية التى يراها مناسبة حيال الإحلال والمحددة على النظام من جملة أربع إستراتيجيات هى:

1 - FIRST IN FIRST OUT (FIFO).
2 - LEAST RECENTLY USED (LRU).
3 - MOST RECENTLY USED (MRU).
4 - LEAST FREQUENTLY USED (LFU).

ويتطلب البديل الأول أن يسجل نظام التشغيل ميعاد (توقيت) دخول كل صفحة إلى الذاكرة ويتم إحلال الأقدم تواجداً دون إعتبار لشدة الطلب عليها وهذا فى حد ذاته يشكل عائقاً أمام تنفيذ إستراتيجية FIFO التنفيذ الأمثل ، كما يتطلب البديل الثانى تحديد وقت كل عملية تجرى على محتوى الصفحة والأقل إحتياجاً تخرج أولاً بينما يتطلب البديل الثالث إحلال الأحدث تشغيلا من خلال بصمة زمنية على الصفحة.

ويطبق الروتين السابق منطقياً على النحو:

إذ (١) لا يوجد إطار خال.

- حينئذ أختار صفحة لإحلالها.
- حدث الجداول.

إذ (٢) حدث تعديل فى محتوى الصفحة.

- حينئذ أكتب الصفحة على القرص .

---

(١) نستخدم PAGE INTERRUPT , PAGE FAULT للدلالة على نفس المفهوم علماً بأن المراجع الأجنبية تستخدم المسمى الأول.

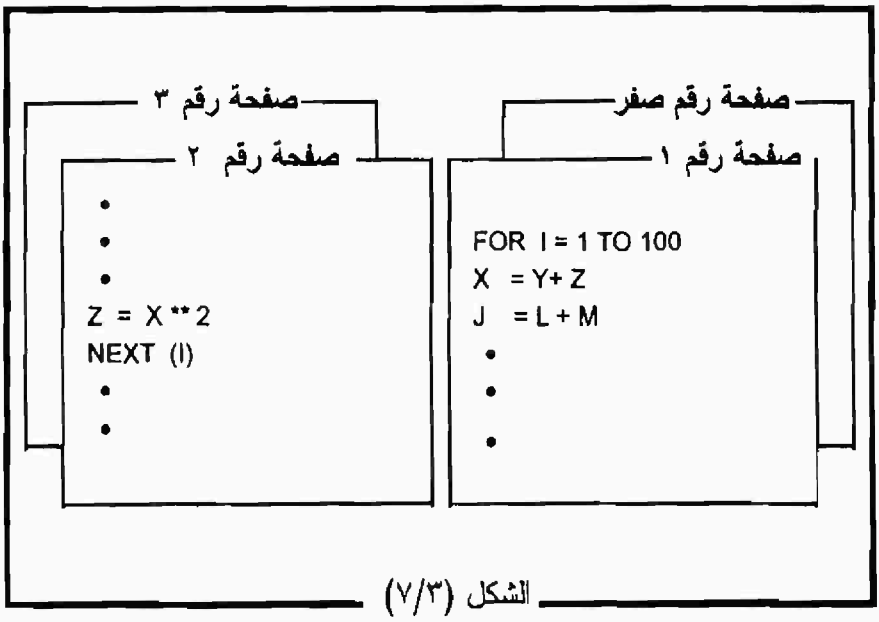
نهاية إذ (٢)

نهاية إذ (١)

## عيوب نظام الصفحات المطلوبة:

رغم أن هذا الأسلوب حل مشكلة عدم توافر حيز فى الذاكرة إلا أن له جملة من المشاكل ومن أبرز مشاكله مشكلة THRASHING فيما يمكن ترجمتـــه [الـــتردد السريع بين الذاكرة الأساسية ووسائط التخزين الثانوية] مما يقلل من كفـــاءة النظـــام والوقت المنتج على وحدة التشغيل المركزية.

وتظهر مشكلة التردد السريع فى تنفيذ دوارة LOOP مقسمة فيزيائيـــاً بيـــن صفحتين إحداهما فى إطار على الذاكرة والأخرى علـــى الأقـــراص علـــى النحـــو الموضح فى الشكل (٧/٣) والمسطر به دوارة مكتوبة بلغة البيزك تتطلـــب إعـــادة وتكرار عملية حسابية مائة مرة بأستخدام الدوارة FOR ..... NEXT .




الشكل (٧/٣)

ومثل هذه الدوارات شائعة الأستخدام فى كل لغات البرمجة ، ففى لغة كوبول يستخدم الأمر PERFORM بصيغة الخمسة عموماً.

وفيما يتم تنفيذ الصفحة رقم (١) [نرمز لها ص ١] حتى أخر إيعاز بعدها يتم إحلالها بواسطة [ص ٢] ويبدأ تنفيذ الإيعاز الأول بها ، لكن عندما يصل التنفيذ إلى الإيعاز NEXT ، يجب إحلال [ص ١] محل [ص ٢] حتى تتم الدوارة عملها ، وبالتالى يجرى هذا الإحلال مائة مرة لإستكمال مخرجات الدوارة ، والحل الوحيد لوقف هذا الإهدار فى الموارد المتاحة وجود إطار خال بحيث تتواجد الصفحتان فى الذاكرة الأساسية فى نفس الوقت ليقل مقاطعات الصفحات PAGE FAULT ويقلل من كثرة أستدعاء روتين PAGE INTERRUPT HANDLE R لهذا يحتاج تطبيق هذا الأسلوب إلى فهم مستخدمى النظام لعدة أمور أبرزها:

أ ــ حيز الصفحة المتاح.

ب ــ خصائص نظام التشغيل الفاعل المستخدم على الحاسب المتاح.

ج ــ تصميم جيد للبرامج للإقلال من عدم تواجد الصفحات المتصلة منطقيا داخل الذاكرة.

## حساب عناوين المواقع فى حالة التصفح:

**مثال (١):**

أحسب عنوان الموقع للبايت ٢١٤ فى نظام تصفح يستخدم ١٠٠ بايت حيزاً للصفحة؟

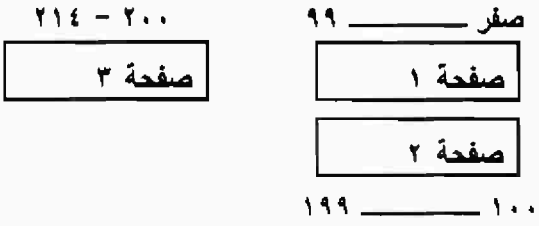
**الحل:**

أقسم عدد البايت على حيز الصفحة والنتيجة كالآتى:

عدد البايت ÷ حيز الصفحة = عدد الصفحات
باقى القسم إن وجد = الإزاحة

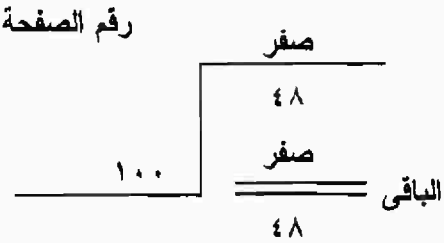
إذا رقم الصفحة = ٢ وفيزيائياً هى الصفحة رقم (٣) حيث الصفحة رقم صفر.

والإزاحة = ١٤




**مثال (٢):**

أحسب عنوان الموقع للبايت ٤٨ فى نظام التصفح السابق ؟




رقم الصفحة = صفر

الإزاحة = ٤٨

الموقع الفعلى = الإطار رقم (١).

بالرجوع إلى جداول أطر الصفحات يمكن تحديد العنوان الحقيقى :

| رقم الصفحة | رقم إطار الصفحة المناظر |
|---|---|
| صفر | ٩ |
| ١ | ١٢ |
| ٢ | ٢٧ |
| ٣ | ٥ |

**ملحوظـــة:**

الصفر ناتج القسمة لا يعنى معناه المجرد "الحسابى" انمـــا يعنـــى أن رقـــم الصفحة هو صفر.. وغير غائب عن البال عدم وجود كسر صفحة.

**مثـــال (٣):**

المطلوب إيجاد إيعاز على الصفحة الثانية بإزاحة مقدارها ١٤؟

يقوم نظام التشغيل بإجراء الحسابات التالية:

رقم الصفحة = خارج القسمة الصحيح للعناوين على حيز الصفحة.

الإزاحة = باقى خارج القسمة السابق.

من الجدول السابق فإن الصفحة رقم (٢) تشغل الإطار رقم ٢٧

أول عنوان فى إطار الصفحة = رقم إطار الصفحة × حيز الصفحة

= ٢٧ × ١٠٠ = ٢٧٠٠

إذا عنوان الإيعاز فى الذاكرة = ٢٧٠٠ + ١٤ = ٢٧١٤

والملاحظ أن هناك عمليات طويلة أخرى لكى يتحول النظام مـــن حســـاب الموقـــع النسبى للإيعاز أو السجل أو البايت إلى حساب عنوان الموقع المطلق على وســـائط التخزين الثانوية وكلها تشكل تكلفة إضافية على موارد النظام وكذلك تستنزف وقتـــاً من جملة الوقت المنتج لوحدة المشغل.

## أسلوب تجزىء الذاكرة SEGMENTATION :

يعتبر هذا الأسلوب مناسبا للبرامج المكتوبة وفق المفهوم الهيكلى حيث يتركب البرنامج من هيكل رئيسى ومجموعة تركيبات صغيرة ويندر أن يكون حيزها متساوياً (وهذا ما يفرق بينه وبين أسلوب التصفيح) ويتم شحن الأجزاء فى الذاكرة وفق مؤشرات POINTERS توضح الإرتباط المنطقى بينها.

ويحتاج برنامج قيادة الذاكرة الأساسية إلى إنشاء مجموعة جداول تسجل كل المعلومات المتاحة عن الأجزاء المختلفة ، وهذه الجداول تجمع بين مفاهيم أسلوب القطاعات الديناميكية [ راجع الباب السادس ] وبين أسلوب الصفحات المطلوبة ، وأبرز هذه الجداول هى:

أ ــ جداول المهام التى يجرى تشغيلها على النظام.

ب ــ جدول يوضح بيانات الأجزاء وإرتباطاتها المنطقية.

ج ــ جدول خريطة الذاكرة محدداً به الحيز المتاح لكل مهمة.

وكما فى أسلوب "الصفحات المطلوبة" فإن تنفيذ الإيعازات يتم تسلسلياً داخل كل الأجزاء والتى لا يشترط وجود تلامس فيزيقى بينها ويحتاج نظام التشغيل بيانات توضح أين يوجد كل جزء على أمتداد حيز الذاكرة الأساسية ، حيث تتم إجراءات مناظرة لما يتم فى أسلوب التصفح مما يستدعى عنونة الأجزاء ورقم كل جزء والإزاحة من بداية الجزء OFFSET-DISPLACEMENT ولأن الأجزاء غير متساوية مما يتطلب أستخدام برامج ضبط لمراجعة العناوين حتى لاتخرج عن حدود الجزء.

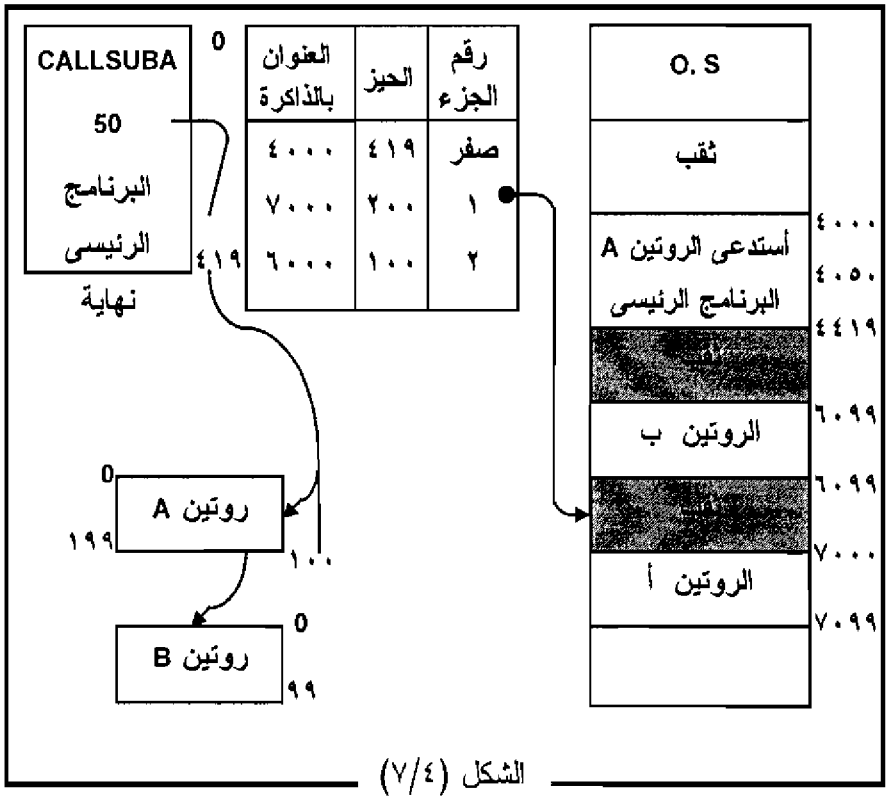
ويوضح الشكل (٧/٤) المفاهيم التى أسلفناها وكذلك حلاً للمثال التالى:

**مثـــال:**

كتب أحد البرامج باسلوب الهيكلى ووجد أنه يتكون من:

* هيكل رئيسى .
* روتين A .
* روتين B .

وبفرض أن الهيكل الرئيسى يستدعى الروتين A عند العنـــوان ٥٠ ، وأن الروتين A يستدعى الروتين B عند العنوان ١٠٠ ، فما هى الإجراءات التى ســـوف تتخذ لأدارة هذا البرنامج وفق أسلوب التجزىء.




الشكل (٧/٤)

**الحـــل:**

يتولى مدير الذاكرة حساب المواقع على النحو التالى بالرجوع إلى الجداول الثلاث لكل جزء.

عناوين الهيكل تبدأ من ٤٠٠٠ وتستمر حتى ٤٤١٩.

عنوان الأستدعاء للروتين أ = ٤٠٠٠ + الإزاحة من بداية القطعة البرمجية

= ٤٠٠٠ + ٥٠ = ٤٠٥٠

وتبدأ تنفيذ الروتين ( أ )

عند العنوان ٧٠٠٠ + ١٠٠ = ٧١٠٠

يستدعى الروتين (B) ويطبق القانون التالى:

Actual Memory Location =
Begining Memory Location of Segment + Displacement

وهو ما يشبه تماما القانون المطبق فى حالة التصفح PAGING .

## أسلوب المزج بين التجزىء والصفحات المطلوبة

**: SEGMENTED / DEMAND PAGED MEMORY ALLOCATION**

وهو عبارة عن مزج بين أسلوبى التجزىء والصفحـــات المطلوبـــة ولـــه المميزات المنطقية لأسلوب التجزئى ، وكذلك المميزات الفيزيائية لأسلوب التصفـــح ونخلص إليه بأن الجزء لايتم شحنه فى الذاكرة الأساسية فى مواقع متصلة لكن يقسم إلى صفحات ذات حيز ثابت ومعلوم لكنها أصغر من أى جزء وبالتالى فهى أســـهل فى المعالجة والتشغيل من أستخدام الجزء كوحدة متكاملة ، وبالتالى فإن كل مشـــاكل أسلوب التجزىء مثل (تكوين الفتات مما يترتب عليه الحاجة إلى إجـــراء عمليـــات التضاغط) لا وجود لها لأن الصفحات لها حيز ثابت وإن لم يلغ ذلك التفتيت الداخلى وإن كانت براعة المبرمج قادرة على تلافيه والإقلال منه للحد الأدنى.

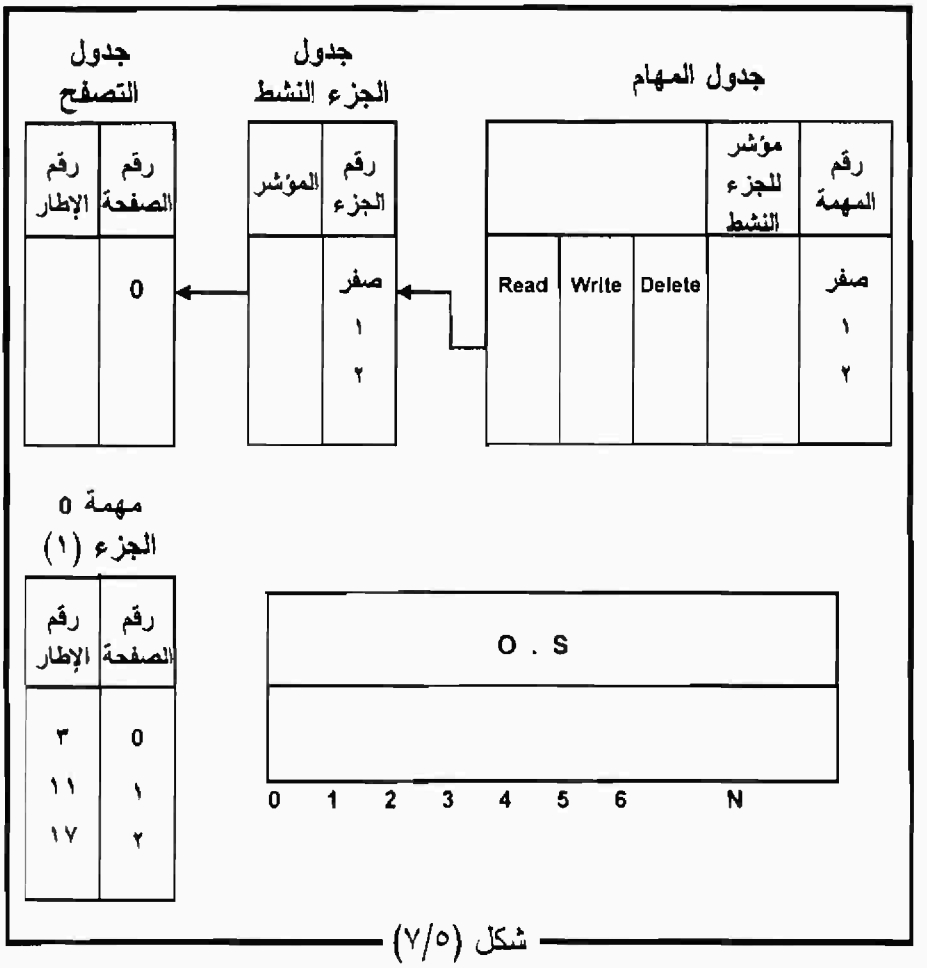
ويتطلب هذا المزج إنشاء أربعة جداول تساند قيادة الذاكرة على النحو التالى والموضح بالشكل (٧/٤):

أ ــ جدول المهام النشطة.

ب ــ جدول الجزء النشط.

ج ــ جدول التصفح.

د ــ جدول مواقع الذاكرة.




شكل (٧/٥)

ويلاحظ أنه للوصول إلى أى موقع فى الذاكرة فإن العنوان يشمل ثلاث بيانات هى (رقم الجزء ــ رقم الصفحة فى نفس الجزء ــ الإزاحة داخل الصفحة) ويضاف عليها مراجعة رقم إطار الصفحة فى حالة تحديد موقعها فى الذاكرة الأساسية.

**عيوب أسلوب التجزىء:**

١ ــ تتطلب تكلفة إضافية فى إنشاء الجداول وصيانتها.

٢ ــ تستقطع زمنا من وحدة التشغيل المركزية فى تشغيل وأدارة الجداول.

**علاج عيوب النظام:**

تستخدم مجموعة من المسجلات فى الذاكرة الأساسية تحفظ فيها [العناوين] بين مجموعة من الأجزاء SEGMENTS وكذلك أرقام الصفحات المنتمية إليها مع عناوينها فى الذاكرة بغرض الربط بين الذاكرة الأساسية والجزء والصفحات.

والمسجلات أماكن فى الذاكرة يتراوح عددها حتى ثمانية مسجلات وقد يختلف العدد وفق البرنامج المستخدم ويسمى هذا الأسلوب الذاكرة المرتبطة ASSOCIATIVE MEMORY .

والسؤال: كيف يعمل النظام وفق هذا الأسلوب؟ والاجابة: عندما تخصص مواقع فى الذاكرة لمهمة يشحن جدول الأجزاء فى الذاكرة بينما لا يشحن جدول الصفحات إلا عند الطلب وبينما يتم تبادل وإحلال الصفحات بين الذاكرة ووسائط التخزين الثانوية فإنه يتم تحديث كل الجداول.

## أسلوب الذاكرة التخيلية VIRTUAL MEMORY:

يقصد بالذاكرة التخيلية القدرة على عنونة مواقع أكبر من الحيز الحقيقى المتاح فى الذاكرة الأساسية، مما يعطى إيحاءاً للمستخدم بأن برنامجه موجود بالكامل فى الذاكرة بينما هو فى الواقع موجود على وسائط التخزين الثانوية المباشرةDASD ولا يتواجد فى الذاكرة الجزء الفاعل أو الصفحات المطلوب تشغيلها لحظة العمل ،

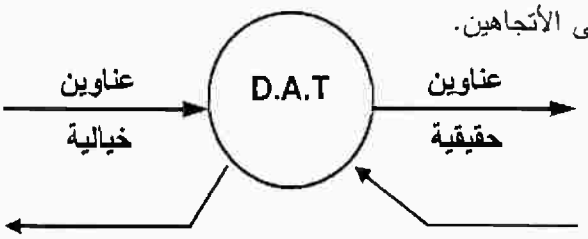
ويعتبر هذا الأسلوب من أنجح الأساليب فى تعددية البرمجة ، ويتم أستخدام هذه التقنية بأحد أسلوبين:

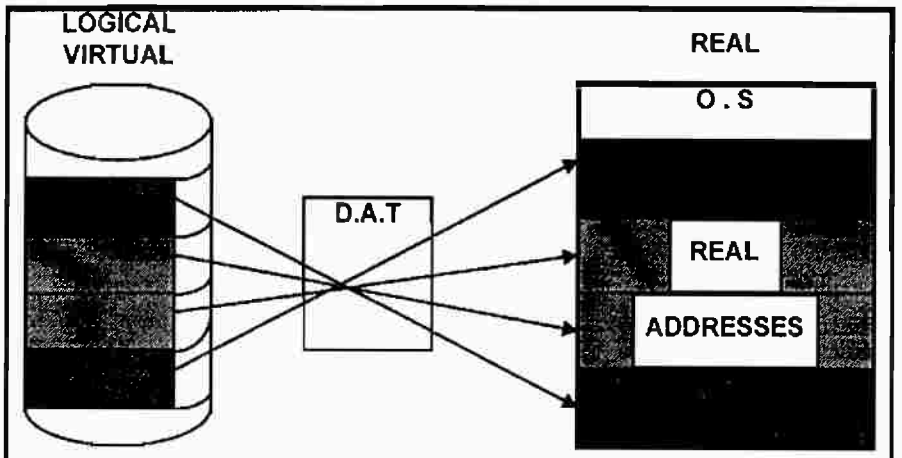
الأول : التصفح PAGING .

الثانى: القطاعات PARTIONING .

وتسمى العناوين التى تخزن وفقها البرامج على وسائط التخزين عناوين خيالية لكن عند التنفيذ لابد أن تتم على عناوين حقيقية ( R ) REAL ، ويقصد بذلك بأن العنوان التخيلى ( V ) VIRTUAL يجب أن يناظره عنوان حقيقى يجرى تخليقه بنفس سرعة التشغيل [التنفيذ] على الحاسب وإلا أنهارت كفاءة النظام وتحطم الأسلوب من أساسه ، ويتم التوفيق والربط بين العنوان التخيلى ( V ) والحقيقى ( R ) بأستخدام نظام آلى للتحويل يسمى المترجم الديناميكى للعناوين Dynamic Address Translator (D.A.T) فيما يوضحه الشكل (٧/٦).

ويعتبر المترجم الديناميكى للعناوين التخيلية إلى عناوين حقيقية أحد أبرز الكيانات فى أسلوب الذاكرة التخيلية ، وله كيانان ، كيان آلى من دوائر إلكترونية بالغة السرعة ، وكيان برمجى يسخر الكيان الآلى للمترجم الديناميكى لحساب العنوان الحقيقى مقابل العنوان الوهمى ، كما يؤدى الوظيفة العكسية ، فبعد إنتهاء معالجة الصفحة يتحول العنوان [الحقيقى ــ الفيزيائى] إلى ذات العنوان الوهمى على الأقراص ، ويساعد المحول الديناميكى إنشاء مجموعة من الجداول تساعده على أداء وظائفه فى الأتجاهين.






الشكل (٧/٦)

وتتم ترجمة العناوين بسرعة عالية جداً تناسب سرعة تشغيل الصفحات وأنتقالها من V إلى R والعكس وعلى المترجم الديناميكى للعناوين يتوقف نجاح هذا الأسلوب.

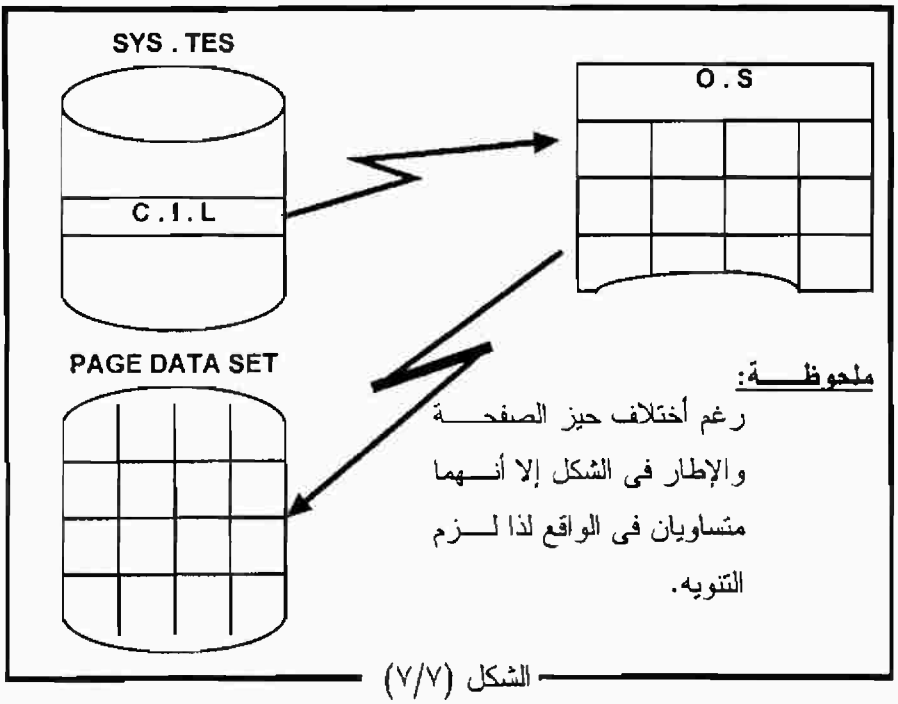
وإتاحة هذا الأسلوب إضافة إلى تعددية البرمجة ترتب عليها تعظيم عدد مستخدمى النظام مما يستدعى أستخدام وسائط تخزين ثانوية عالية السرعة DASD ذات حيز تخزين ضخم ، كما يتطلب استخدام نظام المترجم الآلى فى تسجيل كل المعلومات عن العناوين خلال مراحل تنفيذه مستخدماً فى ذلك عدداً من جداول النظام.

## خطوات تنفيذ برنامج:

توضح الأشكال (٧/٨،٧) خطوات التنفيذ وهى على النحو التالى:

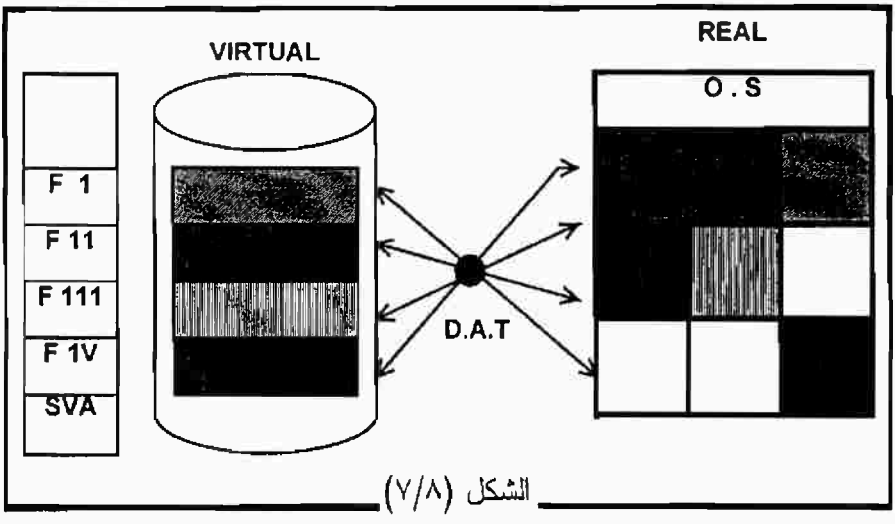
أ ــ إذا كان هناك أطر صفحات خالية تنتقل الصفحات من ذاكرة القلب المصورة C . I . L إلى العناوين R فى الذاكرة الأساسية فإذا لم تتوفر الأطر فانه يتم إحلال صفحات إلى وحدة DASD حتى تتواجد صفحات جديدة.

ب ــ بشأن أعادة الكتابة وإحلال الصفحات يتبع ما جاء فى التصفح.




الشكل (٧/٧)

ج ــ فى حالة تعددية البرامج: يجرى توصيف القطاعات التخيليـــة ( V ) علـــى وحدات DASD.




الشكل (٧/٨)

**مميزات هذا الأسلوب:**

أ ــ يسمح بأى حجم للبرنامج الواحد.

ب ــ أستخدام جيد للذاكرة حيث تستغل فى ادارة أكثر من برنـــامج ودون حدود.

ج ــ تلغى تماماً التفتيت الخارجى وتقلل لأدنى حد من التفتيت الداخلى.

د ــ تسمح بالمشاركة فى البيانات.

هـ ــ تسمح بالربط الديناميكى بين أجزاء البرامج.

**العيـــــوب:**

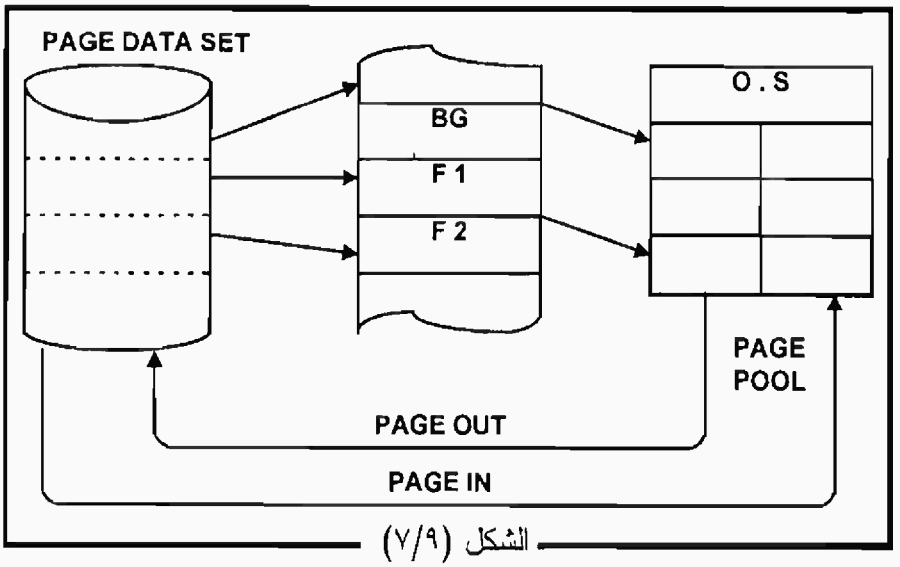
أ ــ يتطلب معدات إضافية.

ب ــ تكلفة إضافية لمعالجة مقاطعات الصفحات PAGE INTERRUPTS .

ج ــ زيادة تعقيد البرامج لمنع THRASHING .

**مسميات شركة IBM :**

يعرض الشكل (٧/٩) مسميات IBM المناظرة :




الشكل (٧/٩)

**أسلوب فئة الصفحات المؤثرة WORKING PAGE SET :**

**وتسمى أحياناً ACTIVE PAGE SET:**

الصفحات المؤثرة.. هى عبارة عن مجموعة الصفحات التى تؤثر على تنفيذ البرنامج والتى يجب أن تتواجد بأستمرار فى الذاكرة الأساسية وتبقى الأخريات غير المؤثرات على وسائط التخزين الثانوية ، مما يوفر أطر صفحات يمكنها أستقبال صفحات مؤثرة من برنامج آخر مما سوف يعظم عدد البرامج المنفذة على المنظومة.

ويعتبر تحديد الصفحات المؤثرة من أصعب المهام التى تواجه مصمم البرنامج خاصة فى النظم التى يتم انشاؤها حديثا حيث يصعب التحديد المسبق للصفحات المؤثرة حتى لو أستخدم الحدس ومهما كانت الخبرة.

## حيز الصفحة:

بعد عرض مختلف أساليب أدارة الذاكرة بالتصفح فإن الهاجس المنطقى.. ما هو الحيز المثالى للصفحة؟ وهل يتحتم بالضرورة أن يكون حيز الصفحة نمطياً لجميع نظم التصفح فى مختلف نظم التشغيل؟ وما القيود على أستخدام صفحات ذات حيز أكبر؟ وهل الصفحات ذات الحيز الأكبر يجب أن تكون مضاعفات صحيحة من أصغر حيز ؟

والإجابة على هذه الأسئلة تتطلب فهماً عميقاً للكيان البرمجى للحاسب SOFT WARE والكيان الآلى للحاسب HARD WARE وكذلك التطبيقات الجارية على نفس الحاسب.. معنى هذا أنه لا توجد إجابة شافية أو بمعنى أدق إجابة ترد على كل تلك الأسئلة بصورة محددة يمكن تعميمها على كل النظم التى تستخدم أسلوب الذاكرة التخيلية تحت أى مسميات للشركات.

والآن نناقش إنعكاسات تصغير حيز الصفحة:

أ ـ كلما تقلص حيز الصفحة زاد عددها طردياً مما يستدعى أستخدام جداول / عنونة الصفحات / ممتدة ، وفى بعض النظم تحتل هذه الجداول أماكن فى الذاكرة الأساسية تشغل حيزاً كبيراً يؤثر على حركة تبادل الصفحات ويؤثر على أداء النظام بصورة عامة ، وهذا يؤكد الحاجة إلى صفحات ذات حيز أكبر.

ب ـ عند استخدام صفحات كبيرة الحيز سوف تحتوى على كم كبير من البيانات والتى قد لا تستخدم فى التشغيل ، وهذا يشير إلى الحاجة إلى صفحات تتصف بالحيز المحدود.

ج ـ ونظراً لأن إتمام I/O من على الأقراص مستهلك للوقت فإنه من الأفضل تقليل عدد مرات I/O قدر الامكان ، وهذا يشير إلى الحاجة إلى صفحات كبيرة نوعاً.

د ـ من دراسات على الـ LOCALITY ثبت أنه من أفضل استخدام صفحات ذات حيز صغير مما يساعد على تشكيل فئة صفحات مؤثرة / نشطة / عددها أكبر.

هـ ـ كل هذه الإعتبارات جعلت الإختيار ينصب على صفحات ذات حيز صغير ١٠٢٤ بايت ومضاعفاتها حتى ٤٠٩٦ بايت وهناك حاسبات مثل DEC المزودة بنظام تشغيل [VAX .11] تستخدم صفحة من ٥١٢ حرف.

## مفاهيم أساسية:

### ١ ـ PAGE FAULT مقاطعة الصفحة:

عبارة عن مقاطعة من الكيان الآلى بسبب عدم وجود صفحة فى الذاكرة الأساسية ، وبالتالى يتم إحلال صفحة من الذاكرة الأساسية إلى وسائط التخزين الثانوية ونقل صفحة جديدة إلى الذاكرة الأساسية.

٢ ـ برنامج التعامل مع مقاطعة الصفحات

PAGE INTERRUPT HANDLER :

عبارة عن روتين فى برنامج قيادة الذاكرة يحدد هل هناك إطار صفحة خال وبالتالى يمكن نسخ الصفحة المطلوبة بها من وسائط التخزين الثانوية أو تحديد ما هى الصفحة التى يمكن إزاحتها من إطارها فى حالة اشغال جميع أطر الصفحات المتاحة.

٣ ـ جدول تخريط الذاكرة PAGE MAP TABLE:

جدول حيوى جداً يوجد بالذاكرة الأساسية دوما ويشمل بيانات عن أرقام الصفحات وكذلك أطر الصفحات المخصصة لها.

٤ ـ الإزاحة OFFSET :

ويستخدم فى حالة الصفحات أو التجزىء وهو عبارة عن الفرق بين عنوان الصفحة وموقع السجل [السطر ـ الكلمة ـ الحرف] فى الصفحة.

٥ ـ LOCALITY الإقليمية ـ المحلية:

فى معظم سياسات أدارة الذاكرة الأساسية فإن معظم المهام تميل إلى التعامل REFERENCE مع الذاكرة فى مواطن غير منتظمة قريبة جداً إلى بعضها البعض.

وكما سبق ووضحنا فى باب سابق أنها أحد الأسباب التى دعت إلى صناعة الذاكرة المختفية ومع هذا ظاهرة محسوسة فى المقام الأول أكثر منها مبنية على أساس علمى ، وعلى سبيل المثال ففى بعض نظم التصفيح تلاحظ أن المهام خلال

تنفيذها تميل إلى التعامل مع بعض فئات من الصفحات وأن هذه الصفحات تميل إلى تنسيق نفسها متجاورة فى الذاكرة التخيلية ، وهذا لا يعنى عدم إنجاز المهمة والتعامل مع صفحات البرنامج وإلا لما بدأ التنفيذ من البداية ، لكنها تعنى فى المقام الأول أن المهمة تميل إلى تركيز تعاملها فى فترة زمنية مع هذه الصفحات ، وينبه هذا إلى أن المهمة يتم تنفيذها بكفاءة عالية طالما تطلب تشغيل هذه الفئة من الصفحات.

*