

المحاضرة الثامنة الذاكرة الافتراضية (الظاهرية) Virtual Memory

مقدمة:

في الخمسينيات وقبل ظهور الذاكرة الظاهرية، كان كل برنامج نريد تنفيذه لابد من أن يكون في الذاكرة أولاً، هذا يعني أن أي برنامج يجب أن يكون أقل أو يساوي حجم الذاكرة المتوفرة (الفارغة). تعتبر هذه مشكلة في البرامج الكبيرة، إذ لا يمكن تشغيل برامج حجمها أكبر من حجم الذاكرة. هذا يعني أننا لا نستطيع تشغيل برامج كثيرة بالذاكرة. البرامج كبيرة الحجم يجب تقسيمها إلى برامج صغيرة وتنفيذ كل برنامج على حدة، إذ لا يمكن تحميل أكثر من برنامج بالذاكرة. تشغيله برنامج كبير فلابد من شراء ذاكرة تتسع له، ولكن:

- ستكون مكلفة.
- مهما كانت الذاكرة كبيرة ستكون هنالك برامج أكبر.

تعريف الذاكرة الظاهرية:

نظم التشغيل الحديثة تستخدم جزء من القرص الصلب كامتداد للذاكرة الرئيسية. حيث يحمل جزء من البرنامج في الذاكرة الرئيسية (إذا لم يكن هنالك متسع لتحميله كاملاً في الذاكرة الرئيسية) وباقي أجزاء البرنامج تحمل في جزء من القرص الصلب. في هذه الحالة يعتبر هذا الجزء من القرص الصلب امتداداً للذاكرة الرئيسية ويسمى الذاكرة الظاهرية وبهذا تكون هذه النظم قد حلت مشكلة ارتباط حجم البرنامج بحجم المساحة المتوفرة من الذاكرة الرئيسية.

يبدأ المعالج في تنفيذ جزء البرنامج الموجود بالذاكرة، وإذا أحتاج إلى أوامر أو بيانات من الجزء الآخر (الموجود بالقرص) سيقوم مدير الذاكرة بتبديل الجزئين (swap).

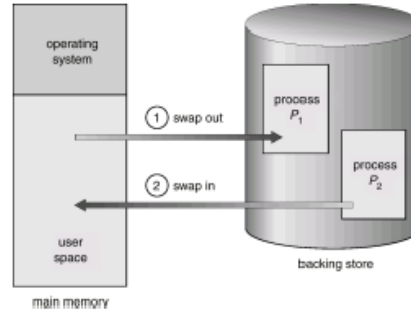
تعتبر الذاكرة الظاهرية فصل بين ذاكرة المستخدم المنطقية والذاكرة الحقيقية وتختلف عن الذاكرة الحقيقية في الآتي:

- فقط جزء من البرنامج يكون بالذاكرة.
- قد يكون مجال العنوان المنطقي أكبر من مجال العنوان الحقيقي.
- يمكن لعدة عمليات التشارك في مجال عناوين.

المبادلة Swapping:

المبادلة هي تحويل البرنامج (أو جزء منه) من الذاكرة الرئيسية إلى القرص أو العكس. تتم المبادلة على مرحلتين، الشكل (9-1)، هما:

- تحويل البرنامج (أو جزء منه) من الذاكرة إلى القرص (swap out).
- تحميل البرنامج (أو جزء منه) من القرص إلى الذاكرة (swap in).



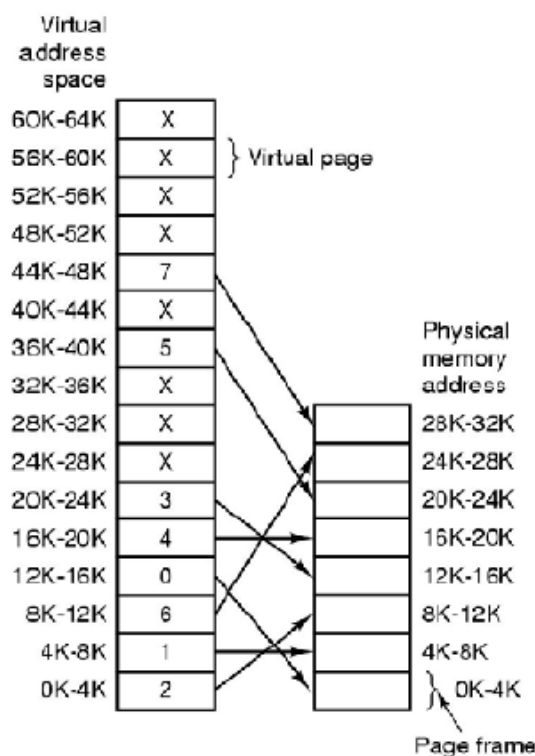
شكل رقم (9-1): المبادلة.

يمكن تطبيق الذاكرة الظاهرية في:

1. إدارة الذاكرة بالصفحات.
2. إدارة الذاكرة بالمقاطع.

الذاكرة الظاهرية في الصفحات:

إذا كان صفحات البرنامج أقل أو تساوي الإطارات بالذاكرة فلن نحتاج إلى ذاكرة ظاهرية. ولكن نحتاج للذاكرة الظاهرية إذا كانت صفحات البرنامج أكبر من الإطارات المتوفرة بالذاكرة. مثلا في الشكل (9-2) نجد أن الذاكرة مقسمة إلى 8 إطارات، والبرنامج مقسم إلى 16 صفحة. سيكون هنالك 8 صفحات بالذاكرة من البرنامج (أمامها علامة X) والبقية ستكون في القرص الصلب (ذاكرة ظاهرية).



شكل رقم (9-2): الصفحات.

التعامل مع الذاكرة الظاهرية:

إحضار الصفحة التي نحتاجها إلى الذاكرة:

• نبحث عن الصفحة المطلوبة في جدول الصفحات

○ إذا كانت الصفحة غير صحيحة، نتوقف

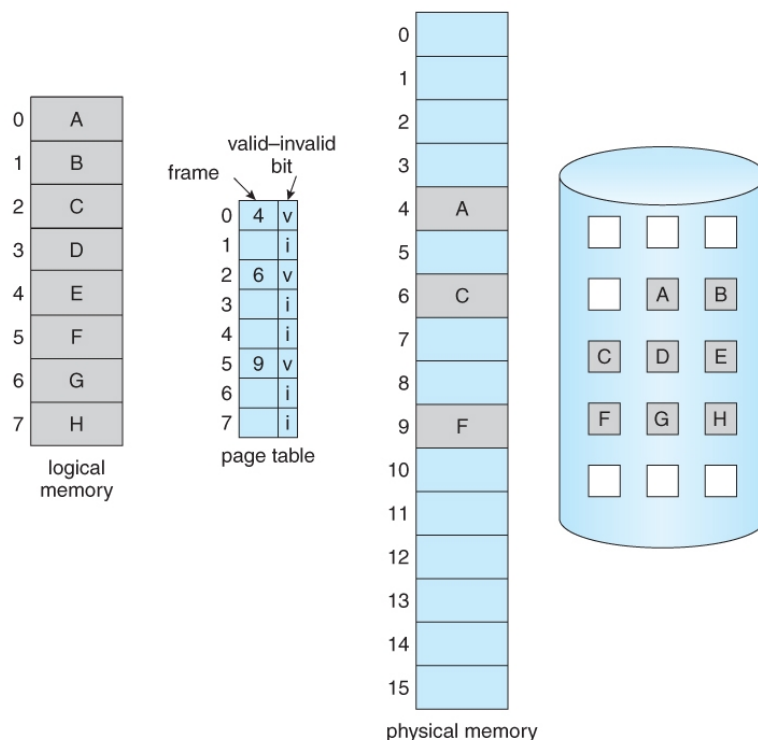
○ إذا كانت ليست بالذاكرة، نحضرها إلى الذاكرة

• لا نحضر صفحة إلى الذاكرة ما لم نحتاجها

هنالك خوارزميات عديدة يستخدمها مدير الذاكرة ليحدد أي صفحة يخرج من

الذاكرة عند ما نريد إطار فارغ لتخزين الصفحة المطلوبة به (فن الصفحة الضحية)؟

قبل إخراج الصفحة (استخدام مكانها)، سيقوم مدير الذاكرة باختبار هل الصفحة المراد استخدامها قد تم تعديلها أم لا ؟ إذا تم تعديلها يجب أن تحفظ بالقرص، أما إذا لم يتم تعديلها فنكتفي بنسختها القديمة الموجودة بالقرص.



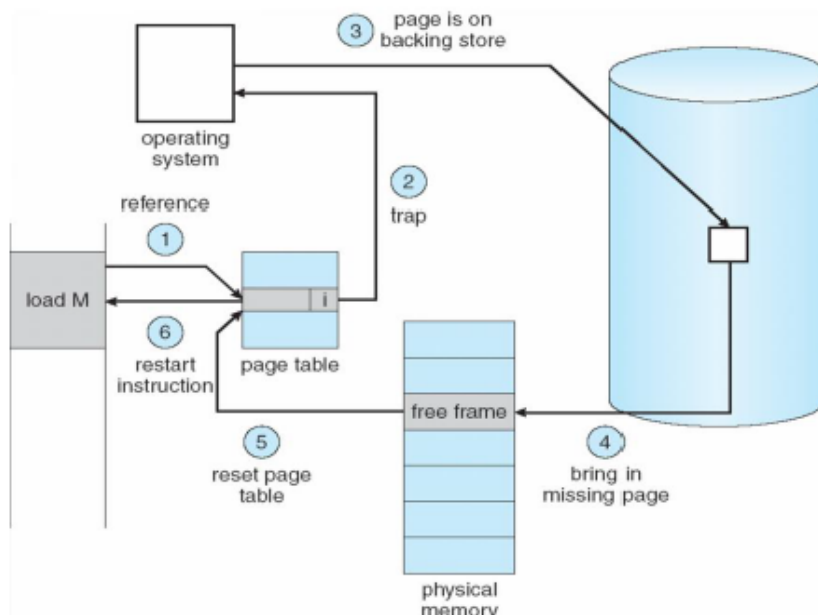
خطأ صفحة Page Fault

عندما يطلب برنامج صفحة غير موجودة بالذاكرة (إفتقار صفحة)، سيسبب ذلك قفز إلى نظام التشغيل برسالة تقول أن هنالك خطأ صفحة page fault ، أي أن البرنامج يريد صفحة ولم يجدها بالذاكرة. على نظام التشغيل توفير الصفحة المفقدة للبرنامج الذي إفتقدها حسب الخطوات التالية:

- إذا كانت الصفحة ليست صحيحة، يتوقف abort.
- إذا كانت الصفحة صحيحة لكنها ليست بالذاكرة:
 - الحصول على إطار فارغ.
 - وضع الصفحة المطلوبة في هذا الإطار الفارغ.
 - تحديث جداول الصفحات بوضع رقم الصفحة ورقم الإطار الذي وضع بها.

○ تعديل بت التصحيح إلى v ، وهذا يعني أن الصفحة صحيحة.

● إعادة تنفيذ الأمر الذي سبب خطأ صفحة - page fault.



ماذا يحدث إذا لم نجد إطار فارغ ؟

إذا طلبنا صفحة غير موجودة بالذاكرة ولم يجد نظام التشغيل إطار فارغ لإحضار الصفحة المطلوبة فيه، ماذا يفعل نظام التشغيل ؟

سيقوم بعملية استبدال (swap)، أي إخراج صفحة من الذاكرة (نسميها الصفحة الضحية) وإدخال الصفحة المطلوبة مكانها.

هنا يظهر سؤال هام وهو أي صفحة سيخرج من الصفحات التي بالذاكرة (من ستكون الصفحة الضحية) ؟ هنالك عدة خوارزميات تحدد أي صفحة سنخرج.

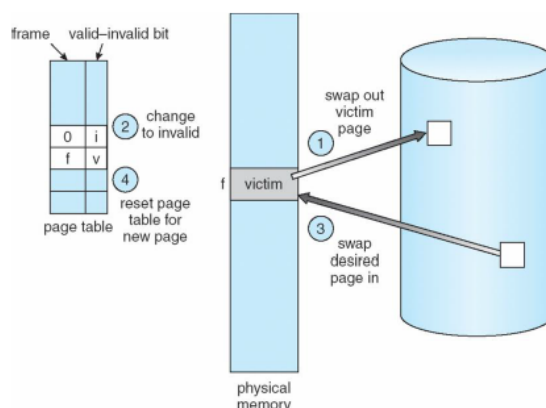
يستخدم نظام التشغيل أحد هذه الخوارزميات، مع وضع إعتبار للأداء في اختيار الخوارزمية (نريد خوارزمية تخرج صفحة لا نحتاجها في القريب العاجل وهذا يقلل عدد أخطاء الصفحات - page faults).

استبدال الصفحات Page Replacement

نستفيد من استبدال الصفحات في استخدام إطارات قليلة لتنفيذ عملية بصفحات كثيرة، وهي الفكرة الرئيسية في الذاكرة الظاهرية.

خطوات استبدال الصفحة:

1. أبحث عن موقع الصفحة في القرص
2. أبحث عن إطار فارغ، إذا وجد إطار نستخدمه.
3. إذا لم يوجد إطار فارغ، نستخدم خوارزمية لإختيار الإطار الضحية **victim** **frame**.
4. إخراج الصفحة الضحية من الإطار الضحية.
5. إحضار الصفحة المطلوبة إلى الإطار الضحية.
6. تعديل جدول الصفحات.
7. إعادة تنفيذ الأمر الذي سبب خطأ الصفحة.



خوارزميات استبدال الصفحات

هنالك العديد من الخوارزميات التي تستخدم في استبدال الصفحات منها:

- إخراج الصفحة الأقدم (الأول أولا تخرج أولا)، first come first out (fifo)
- إخراج الصفحة التي لا نحتاج لها قريبا (الخوارزمية المثلى optimal)
- خوارزمية الصفحة الأقل استخدام LRU
- خوارزمية LRU التقريبية approximation
- خوارزمية الساعة.
- خوارزمية الفرصة الثانية.
- خوارزمية التعداد
- نريد أقل معدل page-fault

سنتحدث عن طريقة عمل كل خوارزمية وسيكون هنالك مثال لكل طريقة، وسنقيم كل الخوارزمية بحساب عدد فقدانها للصفحات - page faults التي قد تحدث على سلسلة معينة طلبات الصفحات.

وتكون سلسلة طلبات الصفحات هي كتابة أرقام الصفحات حسب ترتيب طلبها، مثلا السلسلة التالية:

1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5

توضح أننا نريد الصفحة رقم 1، ثم رقم 2، ثم رقم 3، وهكذا.

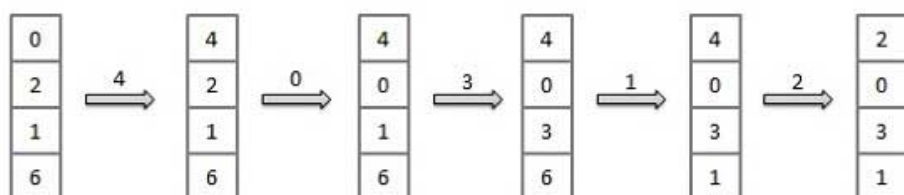
1. خوارزمية الصفحة الأقدم تحميلا FIFO

استبدال الصفحة الأقدم (أقدم صفحة تم تحميلها إلى بالذاكرة .) أي التعامل مع الصفحات الموجودة بالذاكرة حسب زمن وصولها، فالتى تصل إلى الذاكرة أولا تخرج من الذاكرة أولا . عيب هذه الخوارزمية أنه قد يكون لدينا صفحات في الذاكرة غير مستخدمة لمدة طويلة.

- Oldest page in main memory is the one which will be selected for replacement.
- Easy to implement, keep a list, replace pages from the tail and add new pages at the head.

Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1

Misses : x x x x x x x x x



$$\text{Fault Rate} = 9 / 12 = 0.75$$

مثال (6-9)

إذا كان لدي سلسلة طلبات الصفحات التالية:

1,2,3,2,1,5,2,1,6,2,5,6,3,1,3,6,1,2,4,3

ما هي عدد خطأ الصفحات (page fault) إذا استخدمنا 3 إطارات فارغة.

الحل

في البداية ستكون الإطارات الثلاثة فارغة، سنحتاج أولاً للصفحة رقم 1 ولأنها غير موجودة (أول خطأ صفحة) سنحضرها في إطار من الثلاثة، بعدها سنحتاج الصفحة رقم 2 (غير موجودة مما سيسبب خطأ صفحة ثاني) سنحضرها في الإطار الثاني، بعدها سنحتاج إلى الصفحة رقم 3 (غير موجودة أيضاً لذلك سنحضرها إلى الإطار الثالث). الآن عدد خطأ الصفحات هو 3، والإطارات الثلاثة ممتلئة.

سنحتاج بعدها الصفحة رقم 2 والصفحة رقم 1 (كلها موجودة بالذاكرة لا يوجد خطأ صفحة). بعدها سنحتاج الصفحة رقم 5 وهي غير موجودة بالذاكرة لذلك سبب ذلك خطأ صفحة، هذا بالإضافة إلى أننا نريد استبدال صفحة لأن الإطارات الثلاث ممتلئة، لذلك سنخرج الصفحة الأقدم وهي الصفحة رقم 1 وندخل مكانها الصفحة رقم 5، الآن عدد خطأ الصفحات أصبح 4. سنحتاج الصفحة رقم 2 وهي موجودة، بعدها سنحتاج الصفحة رقم 1، ولأنها غير موجودة فسيكون هناك خطأ صفحة (أصبح عدد أخطاء الصفحات 6)، وسنستبدل الصفحة 2 مكان الصفحة 1 لأنها الأقدم. وهكذا ستتم عمليات الاستبدال إلى نهاية السلسلة كما في الشكل التالي بحصيلة عدد 14 خطأ صفحة.

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
1	1	1				2		3	5	1	6		2	5	3		1	6	2
		2	2			3		5	1	6	2		5	3	1		6	2	4
		3				5		1	6	2	5		3	1	6		2	4	3

Example-1 Consider page reference string 1, 3, 0, 3, 5, 6 with 3 page frames. Find number of page faults.

Page reference		1, 3, 0, 3, 5, 6, 3					
1	3	0	3	5	6	3	
		0	0	0	0	3	
	3	3	3	3	6	6	
1	1	1	1	5	5	5	
Miss	Miss	Miss	Hit	Miss	Miss	Miss	
Total Page Fault = 6							

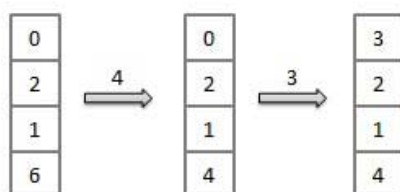
2. الخوارزمية المثلى Optimal Page algorithm

استبدال الصفحة التي لن نطلبها قريباً، أي الأبعد في السلسلة.

- An optimal page-replacement algorithm has the lowest page-fault rate of all algorithms. An optimal page-replacement algorithm exists, and has been called OPT or MIN.
- Replace the page that will not be used for the longest period of time. Use the time when a page is to be used.

Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1

Misses : x x x x x x



$$\text{Fault Rate} = 6 / 12 = 0.50$$

مثال (7-9)

إذا كان لدي سلسلة طلبات الصفحات التالية:

1,2,3,2,1,5,2,1,6,2,5,6,3,1,3,6,1,2,4,3

ما هي عدد أخطاء الصفحات (page fault) إذا استخدمنا 3 إطارات فارغة.

الحل

في البداية ستكون الإطارات الثلاثة فارغة، سنحضر الصفحات 1، 2، 3 في الإطارات الثلاث الفارغة مما يولد 3 خطأ صفحة. بعدها سنحتاج إلى الصفحات 1، 2، 5، 6، 3، 1، 3، 6، 1، 2، 4، 3. وهما بالذاكرة فنستخدمهما دون حدوث خطأ صفحة. بعدها سنحتاج الصفحة 5، وهي ليست بالذاكرة لذلك لابد من إخراج صفحة وإدخالها مكانها. سنخرج الصفحة التي لا نطلبها قريباً (لدينا الآن بالذاكرة الصفحات 1، 2، 3) فإيها سنخرج:

- الصفحة 1 سنطلبها بعد صفحتين من الصفحة 5
- الصفحة 2 سنطلبها بعد الصفحة 5 مباشرة.
- الصفحة 3 سنطلبها بعد 6 صفحات، وهي الأبعد لذلك سنخرجها ونضع 5 مكانها، الآن لدينا بالذاكرة الصفحات 1، 2، 5.

سنحتاج الصفحة 2 وهي موجودة بالذاكرة، وبعدها سنحتاج الصفحة 1 وهي بالذاكرة، بعدها سنحتاج الصفحة 6 التي لا توجد بالذاكرة لذلك سنبدلها بالصفحة 1 لأنها الأبعد. وهكذا إلى أن نصل للحل التالي:

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
1	1	1			1			6			6	6				2	2		
		2	2		2			2			2	1				1	4		
			3		5			5			3	3				3	3		

سيكون لدينا في النهاية 9 خطأ صفحة.

تمرين:

Example-2: Consider the page references 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, with 4 page frame. Find number of page fault.

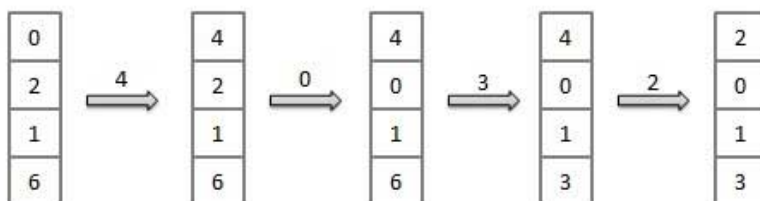
3. التي لم تستخدم حديثاً (LRU) Least Recently Used

بافتراض أن الصفحة التي استخدمت حديثاً هي التي سنحتاجها وهي التي ستستخدم مرة ثانية، أما التي لم تستخدم حديثاً فغالبا لن نحتاجها في القريب العاجل. لذلك سنخرج الصفحة التي لم تستخدم حديثاً (الأقدم استخداماً).

- Page which has not been used for the longest time in main memory is the one which will be selected for replacement.
- Easy to implement, keep a list, replace pages by looking back into time.

Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1

Misses : x x x x x x x x



$$\text{Fault Rate} = 8 / 12 = 0.67$$

مثال (8-9)

إذا كان لدي سلسلة طلبات الصفحات التالية:

1,2,3,2,1,5,2,1,6,2,5,6,3,1,3,6,1,2,4,3

ما هي عدد أخطاء الصفحات (page fault) إذا استخدمنا 3 إطارات فارغة.

الحل

في البداية ستكون الإطارات الثلاثة فارغة، سنحضر الصفحات 1، 2، 3 في الإطارات الثلاث الفارغة مما يولد 3 خطأ صفحة. بعدها سنحتاج إلى الصفحات 2، 1 وهما بالذاكرة فنستخدمهما دون حدوث خطأ صفحة. بعدها سنحتاج الصفحة 5، وهي ليست بالذاكرة لذلك لابد من إخراج صفحة وإدخالها مكانها. سنخرج الصفحة الأقل استخداماً (لدينا الآن بالذاكرة الصفحات 1، 2، 3) فإيها سنخرج:

- الصفحة 1 استخدمت آخر شيء.

- الصفحة 2 استخدمت قبل الصفحة 1.
- الصفحة 3 استخدمت قبل الصفحة 2، أي هي الأول استخداما بين الصفحات الثلاث التي بالذاكرة، لذلك سنخرجها ونضع 5 مكانها، الآن لدينا بالذاكرة الصفحات 1، 2، 5.
- سنحتاج الصفحة 2 وهي موجودة بالذاكرة، وبعدها سنحتاج الصفحة 1 وهي بالذاكرة، بعدها سنحتاج الصفحة 6 وهي ليست بالذاكرة لذلك سنبدلها بالصفحة 5 لأنها الأقل استخداما. وهكذا إلى أن نصل للحل التالي:

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
1	1	1			2			2		6		5	6				6	1	2
		2	2		1			1		2		6	3				1	2	4
			3		5			6		5		3	1				2	4	3

تمرين:

Example-3 Consider the page reference string 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2 with 4 page frames. Find number of page faults.

- تمرين: استخدم خوارزميات LRU , Optimal , FIFO واحسب خطأ الصفحة: قارن النتائج:
- Reference string: 7,0,1,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1
 - 3 frames (3 pages can be in memory at a time per process)
