

10. Memoria Virtuale. Esercizi

- (es. 1) Consideriamo un processo con m frame inizialmente vuoti. La stringa di riferimento è lunga p e contiene riferimenti a n pagine diverse. Per un qualsiasi algoritmo di rimpiazzamento:

- qual è il numero minimo di page fault? n
- qual è il numero massimo di page fault? p

10. Memoria Virtuale. Esercizi

- (es. 2) un computer ha uno spazio degli indirizzi logici di 2^{32} byte. Il computer ha 2^{18} byte di memoria fisica. La memoria virtuale è paginata, con pagine di 4096 byte. Un processo genera l'indirizzo 11123456 (esadecimale). Come viene generato il corrispondente indirizzo fisico?

11123456
 ↳ offset → xy | 456
 ↳ Page Table → xz

(con x non 3)

NON PAGE FAULT

10. Memoria Virtuale. Esercizi

- (es. 3) In un sistema con demand paging, la PT è tenuta in appositi registri. Ci vogliono 8 millisecc per gestire un page fault nel caso migliore, e 20 millisecc nel caso peggiore. L'accesso in RAM richiede 100 nanosecc.
- Una pagina vittima ha il dirty bit a 1 nel 70% dei casi. Qual è la frequenza massima accettabile di page fault per avere un eat massimo di 200 nanosecondi? (esprimete i tempi in microsec.)

no
PT
time

$$0,0002 = (1-p)0,0001 + p[0,7 \cdot 20 + 0,3 \cdot 8]$$

$$0,0002 = 0,0001 - 0,0001p + 16,4p$$

$$0,0001$$

$$\frac{0,0001}{16,4 - 0,0001} = p = 6,1 \cdot 10^{-8}$$

1 ogni

154000

10. Memoria Virtuale. Esercizi

- (es. 4) quali di questi algoritmi: *Optimal*, *FIFO*, *Seconda chance*, *LRU*, soffrono della anomalia di belady?

FIFO	Si
Optimal	no
2 ^a chance	no Si
LRU	no

10. Memoria Virtuale. Esercizi

- (es. 5) quali sono i vantaggi della memoria virtuale? Quali gli svantaggi?

- necessita supporto hardware
 + implementa protezione memoria
 = può rallentare sistema
 + permette di running programmi
 con documenti grandi molto

10. Memoria Virtuale. Esercizi

- (es. 6) In un sistema con demand paging vengono fatte le seguenti rilevazioni sulla percentuale di utilizzo:
 - Utilizzo della CPU: 20%
 - Disco di swap attivo: 97.7%
 - altri device di I/O: 5%

10. Memoria Virtuale. Esercizi

- (es. 6) Quali delle seguenti azioni possono (ragionevolmente) migliorare l'utilizzo della CPU?
 - Installare una CPU più veloce **NO (non usato)**
 - Aumentare l'area di swap **SI NO** → xche poco hd
 - Aumentare il grado di multiprogrammazione **NO**
 - Diminuire il grado di multiprogrammazione **SI**
 - Installare più RAM **SI**
 - Installare un HD più veloce **NO SI (richiede page fault + hd)**
 - aumentare la dimensione delle pagine **NO**
~ solo su sequenziale

10. Memoria Virtuale. Esercizi

- (es. 7) Supponete una politica di rimpiazzamento in cui a intervalli regolari viene rimossa una pagina se questa non è stata usata dall'intervallo precedente.
- Che vantaggi/svantaggi si hanno rispetto ad una politica che usi LRU o seconda chance chiamati solo in caso di page fault?

+ lo spazio è liberato costantemente
 - deve tornare anche se non necessario
 + a momento ^{page fault} chiamato è più veloce
 - causa + page fault (toglie + page del necessario)

MISS = ?

10. Memoria Virtuale. Esercizi

- (es. 8) In un sistema con demand paging, l'area di swap ha un tempo di accesso e di trasferimento di 5 millisecc. Gli indirizzi sono tradotti mediante una PT in RAM, e l'access time in RAM è di 1 μ sec.
- Con l'uso di una memoria associativa, la traduzione di un indirizzo logico in fisico si riduce ad un solo accesso in RAM, l'hit rate è dell'80%, mentre dei restanti accessi, il 10% (ossia il 2% del totale) causano page fault. Calcolare l'effective access time.

$$\begin{aligned}
 \text{ext} &= 0,8 \cdot 1 \mu\text{s} + 0,08 \cdot 2 \text{ms} \\
 &+ 0,02 \cdot 5,002 \text{ms} \\
 &= 101,2 \mu\text{s}
 \end{aligned}$$

PT = 1 μ s
 5 ms di swap
 0,08 missio + PT + swap

10. Memoria Virtuale. Esercizi

- (es. 9) In un sistema con demand paging il grado di multiprogrammazione è stato fissato a 4. In situazioni diverse, alcune misure hanno dato i seguenti risultati:
 - utilizzo CPU: 13% -- utilizzo del disco: 97% *thrashing NO*
 - utilizzo CPU: 87% -- utilizzo del disco: 3% *uso ottimale NO*
 - utilizzo CPU: 13% -- utilizzo del disco: 3% *under used SI*
- Cosa sta succedendo nei tre casi? Si potrebbe aumentare il grado di multiprogrammazione per migliorare l'uso della CPU?

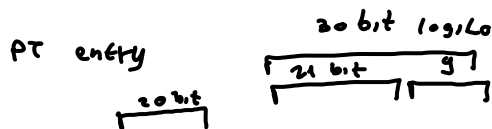
10. Memoria Virtuale. Esercizi

- (es. 10) In un calcolatore la memoria virtuale è di 32K, e la memoria fisica è di 16K. Le pagine del sistema sono di 2K. Si consideri la PT sottostante, e si dia l'indirizzo fisico corrispondente a ciascuno dei seguenti indirizzi virtuali: 565; 4100; 6150 (tutti i numeri sono in base 10)

pagina	frame	valido/inv.
0	3	v
1	6	v
2	x	i
3	2	v
4	x	i

$$\begin{aligned} \text{pagina} &= 2^{11} \\ \text{virtuale} &= 2^{15} \\ \text{fisico} &= 2^{14} \end{aligned}$$

$$\begin{array}{ccc} 565 & 4100 & 6150 \\ \hline 0 \setminus 565 & & \\ \hline 3 \cdot 2048 + 565 & i & 4 \\ \hline = 6709 & \downarrow & \text{invalid} \\ & & \downarrow \\ & & \text{page fault} \end{array}$$



10. Memoria Virtuale. Esercizi

- (es. 11) Un sistema ha le seguenti caratteristiche:
 - È in grado di indirizzare un milione di frames
 - Usa 30 bit per scrivere un indirizzo logico
 - L'offset più grande in un indirizzo è 1FF
- Il sistema dovrebbe prevedere un meccanismo di prevenzione del thrashing? *SI in quanto thrashing*
- Il sistema deve usare una paginazione a più livelli? *SI*
- Nel caso, è sufficiente una paginazione a 2 livelli? *NO*
- Quale dimensione minima dovrebbero avere le pagine per non dover usare una paginazione a più livelli? (per questa domanda si usino 4 byte per ogni entry della PT)

*solo
scemo*

*↓
indirizz
byte non
bit*

$$\begin{aligned} 2) \text{ PT + grande } 2^{21} \text{ entry} \cdot 20 \text{ bit} &= 3 \cdot 2^{21} \text{ byte} \\ &\sim 6,3 \text{ mbyte} > 2^3 \text{ byte} \\ 4) 2^{30-n} \cdot 2^2 \text{ bit} &< 2^n \\ \Rightarrow 30-n+2 &= n \quad 32 < 2n \\ &n > 16 \end{aligned}$$



$$\begin{aligned} n &= 27 \\ 2^{16} \cdot 3 \text{ byte} &> 2^3 \text{ bit} \\ &\downarrow \\ &\text{indirizz } 260 \rightarrow 3 \text{ bit} \\ &\downarrow \\ &4 \\ &\text{indirizz } 23 \text{ bit} \\ &\text{minimo} = 5 \text{ byte} \end{aligned}$$