

memoria virtuale

esercizi

tabella delle pagine (8.1)

- data la tabella delle pagine (pagina di 1024 bytes) di un processo tradurre i seguenti riferimenti in indirizzi fisici: 1052, 2221, 5499

Numero della pagina virtuale	Valid bit	Reference bit	Modify bit	Numero di frame
0	1	1	0	4
1	1	1	1	7
2	0	0	0	-
3	1	0	0	2
4	0	0	0	-
5	1	0	1	0

page replacement (8.4)

Numero della pagina virtuale	Page Frame	Tempo di caricamento	Tempo di riferimento	R bit	M bit
2	0	60	161	0	1
1	1	130	160	0	0
0	2	26	162	1	0
3	3	20	163	1	1

- page fault alla pagina 4 e tempo 164
- quale pagina viene sostituita?
 - rispondi usando FIFO, LRU, Clock (con uso del bit di modifica M), Optimal
 - page reference string: 4 0 0 0 2 4 2 1 0 3 2

working set (8.4)

Numero della pagina virtuale	Page Frame	Tempo di caricamento	Tempo di riferimento	R bit	M bit
2	0	60	161	0	1
1	1	130	160	0	0
0	2	26	162	1	0
3	3	20	163	1	1

- page reference string: 4 0 0 0 2 4 2 1 0 3 2
- $\Delta=4$, variable allocation
- page replacement: LRU
- quanti page fault avvengono? quando?

upper/lower bound (8.12)

- stringa di riferimenti a pagine di lunghezza P
- le pagine distinte nella stringa sono $N \leq P$
- i frames allocati al processo sono M
 - tutti inizialmente vuoti
- algoritmo di replacement non specificato!
- lower bound sul numero di page faults?
- upper bound sul numero di page faults?

anomalia di belady

- considera la seguente stringa di riferimenti
 - 0 1 2 3 0 1 4 0 1 2 3 4
- usa FIFO
- considera 3 frame allocati al processo
 - conta i page fault
- considera 4 frame allocati al processo
 - conta i page fault
- noti qualcosa di strano?

dove mettiamo le page tables?

- su molti sistemi le user page tables di ciascun processo sono parte dell'immagine del processo
- quali sono i vantaggi?
- quali sono gli svantaggi?
 - ciascun riferimento a memoria quanti page fault può generare?

quanti page faults?

- considera la seguente istruzione assembly
mov32 0x00800FFE → 0x00900000
- tale istruzione sposta il contenuto della parola di 4 byte all'indirizzo 0x00800FFE nella parola all'indirizzo 0x00900000
- l'architettura prevede pagine di 4KB e page table a un livello, PTE di 4 byte
 - assumi che le pagine con la tabella delle pagine sia sempre residente
 - indirizzo virtuale: 20 bit page#, 12 bit di offset
- l'istruzione è all'indirizzo 0x00400000 ed è codificata con 6 bytes
- supponi che il PC punti a 0x00400000 e si deve ancora eseguire il fetch
- quanti page fault possono al più aver luogo per l'esecuzione di tale istruzione?

quanti page faults?

- considera la seguente istruzione assembly
mov32 0x00800FFE → 0x008001FFE
- tale istruzione sposta il contenuto della parola di 4 byte all'indirizzo 0x007FFFFE nella parola all'indirizzo 0x008FFFFE
- l'architettura prevede pagine di 4KB e page table a due livelli, PTE di 4 byte, ciascuna pagina contiene 2^{10} PTE
 - assumi che la pagina che contiene la directory sia sempre residente
 - indirizzo virtuale: 10 bit x liv. 1, 10 bit x liv. 2, 12 bit di offset
- l'istruzione è all'indirizzo 0x003FFFFE ed è codificata con 6 bytes
- supponi che il PC punti a 0x003FFFFE e si deve ancora eseguire il fetch
- quanti page fault possono al più aver luogo per l'esecuzione di tale istruzione?