

Programmazione di Sistema (JZ-ZZ) 2022/23

Memory Management

Esercizi proposti tratti da temi di esame

Esercizio 1 (da 05/07/2011)

Sia dato un sistema con memoria fisica di dimensione 512 MB, in cui si utilizza uno schema di gestione a partizioni (contigue) variabili con unità minima di allocazione della memoria di 64 B (ovvero lo spazio di memoria viene allocato in multipli di 64 byte). Al Sistema Operativo sono allocati in modo permanente i primi 128 MB di memoria. La tabella dei processi contiene, per ogni processo attivo, l'indirizzo iniziale (ADDR) e la dimensione (SIZE) della relativa partizione in memoria. La memoria viene allocata con strategia *Worst-Fit*. Le partizioni libere sono gestite mediante una lista linkata ordinata per dimensione decrescente, in cui ogni nodo rappresenta una partizione libera; i nodi della lista sono costituiti da due campi (*puntatore alla partizione successiva* e *dimensioni della partizione*, entrambi rappresentati su 4 byte, dimensioni e indirizzi rappresentati in Byte, con valore 0 usato come puntatore nullo) e sono memorizzati nei primi byte della partizione che rappresentano.

Si supponga che ad un dato istante la tabella dei processi e il puntatore alla prima partizione libera contengano le informazioni rappresentate in figura. Si rappresentino le modifiche alle partizioni in memoria, alla tabella dei processi e alla Free List, in seguito all'attivazione di 2 nuovi processi, P12 e P13, che richiedano rispettivamente 25 MB e 150 MB di memoria, seguita dalla terminazione del processo P11.

Memoria		Tabella dei Processi		
00000000	Sistema operativo	Proc	ADDR	SIZE
08000000	100A0000 08000000
10000000	0B004500 00000000 ...	P10	10000000	000A0000
100A0000	00000000 07760000 ...	P11	17800000	08800000
17800000	00800000 07000000

Free List HDR	
08000000	

Esercizio 2 (da 01/09/2008)

Sia dato un sistema di memoria virtuale con paginazione, nel quale vengono indirizzati i Byte. Il sistema dispone di TLB (*Translation Look-aside Buffer*), su cui si misura sperimentalmente un “hit ratio” del 99 %. La tabella delle pagine (*page table*) viene realizzata con uno schema a due livelli, nel quale un indirizzo logico di 32 bit viene suddiviso (da MSB a LSB) in 3 parti: p1, p2, d, rispettivamente di 10 bit, 11 bit, 11 bit. Non si utilizzano ulteriori strutture dati (quali tabelle di hash o inverted page table) per velocizzare gli accessi.

Si richiede di:

- Dare la definizione di *hit ratio*;
- Illustrare lo schema della page table e la sua dimensione complessiva per un processo P1 avente spazio di indirizzamento virtuale uguale a 100 MB
- Calcolare la frammentazione esterna e interna per il processo P1;
- Calcolare il tempo effettivo di accesso (*EAT*) per il caso proposto (hit ratio = 99%), supponendo un tempo di accesso alla memoria RAM di 300 ns.

Esercizio 3 (da 25/06/2018)

Si descrivano brevemente vantaggi e svantaggi di una inverted page table (*IPT*), rispetto a una tabella delle pagine standard (eventualmente gerarchica). Sia dato un processo avente spazio di indirizzamento virtuale di 32 GB, dotato di 8 GB di RAM, su una architettura a 64 bit (in cui si indirizza il Byte), con gestione della memoria paginata (pagine/frame da 1 KB). Si vogliono confrontare una soluzione basata su tabella delle pagine standard (una tabella per ogni processo) e una basata su IPT. Si calcolino le dimensioni della tabella delle pagine (ad un solo livello) per il processo e della IPT. Si ipotizzi che il *pid* di un processo possa essere rappresentato su 16 bit. Si utilizzino 32 bit per gli indici di pagina e/o di frame. Si dica infine, utilizzando la IPT proposta (32 bit per un indice di pagina/frame), qual è la dimensione massima possibile per lo spazio di indirizzamento virtuale di un processo.