

SISTEMI OPERATIVI E LABORATORIO
(Esonero e Scritto - Indirizzo Sistemi e Reti)
8 settembre 2008

Cognome: _____ **Nome:** _____
Matricola: _____

Ricordate che non potete usare calcolatrici o materiale didattico. Siate sintetici nelle vostre risposte, anche quando è richiesto di motivarle, sono sufficienti poche righe per rispondere correttamente. (Si ricorda che gli studenti degli anni precedenti devono sostenere l'intero scritto).

ESERCIZI RELATIVI ALLA PARTE DI TEORIA DEL CORSO

ESERCIZIO 1 (5 punti)

Quattro processi arrivano al tempo indicato e consumano la quantità di CPU indicata nella tabella sottostante)

Processo	T. di arrivo	Burst
P1	0	10
P2	1	8
P3	1	6
P4	11	3

a) Calcolare il turnaround medio e il waiting time medio per i processi nel caso dell'algoritmo di scheduling SJF preemptive (shortest remaining time first). RIPORTANDO IL DIAGRAMMA DI GANTT USATO PER IL CALCOLO.

Turnaround medio:

(0)...P1...(1)....P3....(7)...P2....(11)...P4....(14)...P2...(18)...P1....(27)
 $P1 = 27$; $P2 = 17$; $P3 = 6$; $P4 = 3$; $53/4 = 13,25$

Waiting time medio:

(basta ricordarsi di sottrarre al turnaround di ogni processo, la durata del suo burst):
 $P1 = 17$; $P2 = 9$; $P3 = 0$; $P4 = 0$; $26/4 = 6,5$

b) Qual è il più grave problema che può avere un algoritmo di scheduling a priorità, e come si risolve?

La possibilità di starvation. Con un meccanismo di aging.

c) Per un sistema time sharing è meglio usare un algoritmo di scheduling preemptive o uno non preemptive? Quale algoritmo suggerite come più adeguato? (motivate la vostra risposta)

E' necessario usare un algoritmo preemptive (ma non SJF, ad esempio, RR, che non produce starvation) in modo da garantire che qualsiasi processo di qualsiasi utente possa prima o poi usare la CPU

d) Riportate il diagramma di stato della vita di un processo.

Vedere figura 3.2, sezione 3.1.2.

e) Come può essere modificato il diagramma di stato della vita di un processo per tener conto di:

1. il processo in esecuzione esegue una fork;
2. il processo in esecuzione esegue una wait su un semaforo la cui variabile semaforica ha valore negativo.

Vedere figura 3.7, sezione 3.2.1

ESERCIZIO 2 (5 punti)

In un sistema con memoria virtuale le pagine sono composte da 10000 (esadecimale) byte, la RAM è fatta di 2^{12} (decimale) frame, e lo spazio di indirizzamento logico massimo è di 256 (decimale) pagine.

a) Qual è la lunghezza in bit di un indirizzo logico? 24 bit ($2^{16} \cdot 2^8 = 2^{24}$)

Qual è la lunghezza in bit di un indirizzo fisico? 28 bit ($2^{16} \cdot 2^{12} = 2^{28}$)

b) Si consideri la PT sottostante (attenzione: nella tabella i numeri sono tutti in base decimale), e si dia l'indirizzo fisico in binario corrispondente al seguente indirizzo virtuale:

113C12: =

n. di pag. = 11 (17 decimale), offset = 3C12 → 080 3C12 = 0000 1000 0000 0011 1100 0001 0010

n. pagina	n. frame	Valido/invalido
0	52	v
1	100	v
2	x	i
3	x	i
4	x	v
5	1	v
6	70	v
7	18	v
8	x	i
9	x	i
10	9	v
11	87	v
12	124	v
13	21	v
14	56	v

15	12	v
16	127	v
17	128	v

c) Si dia un esempio di indirizzo logico generato dalla CPU che, rispetto alla tabella delle pagine riportata, potrebbe essere correttamente tradotto in un corrispondente indirizzo fisico, ma che genera un page fault perché la pagina indirizzata non è caricata in memoria.

030000: = N. di pagina = 03; offset = 0000 → **page fault**

d) Quali sono i principali vantaggi e svantaggi dati dalla memoria virtuale?

Vedere i lucidi della sezione 9.1

e) Che cosa cerca di evitare un algoritmo di rimpiazzamento delle pagine?

Che una pagina selezionata come pagina vittima venga riferita nell'immediato futuro.

ESERCIZIO 3 (4 punti)

a) Un sistema operativo è in grado di decidere, scegliendo tra le tre modalità di base di allocazione dello spazio su disco, quella più adeguata per memorizzare un file in base alle seguenti informazioni, note al S.O. stesso: **(1)** numero di blocchi occupati dal file, **(2)** tipo di accesso al file (*sequenziale* o *diretto*, che viene dichiarato dall'utente all'atto della creazione del file stesso). Per ciascuno dei file riportati qui di seguito, indicate quale modalità di allocazione sceglierà il S.O.:

FileA: 1 blocco, sequenziale: *contigua (o concatenata, l'indicizzata spreca spazio inutilmente)*

FileB: 100 blocchi, diretto: *indicizzata*

FileC: 1 blocco, diretto: *contigua (o concatenata, l'indicizzata spreca spazio inutilmente)*

FileD: 100 blocchi, sequenziale: *concatenata (e' ragionevole anche l'indicizzata, sebbene produca un maggiore spreco di spazio, perché è più affidabile della concatenata)*

b) Nel sistema descritto nel punto a), i blocchi su disco occupano 512 byte, e un puntatore a blocco è scritto su 4 byte. Di un file si sa che deve essere acceduto in modo diretto. Quanti accessi al disco sono necessari per leggere direttamente il contenuto del blocco numero 200 del file, assumendo che gli attributi del file in questione siano già in memoria primaria? (motivate la vostra risposta).

3. Il S.O. userà l'allocazione indicizzata. In un blocco indice possiamo scrivere 128 puntatori, per cui un solo blocco indice non è sufficiente ad indirizzare il blocco 200. Sia usando l'allocazione indicizzata gerarchica che l'allocazione indicizzata concatenata, un secondo blocco indice è sufficiente per indirizzare il blocco 200. Sono quindi necessari due accessi al disco per leggere i due blocchi indice + un accesso per leggere il blocco 200.

c) Consideriamo un generico file all'interno del file system del sistema. Supponiamo per semplicità che nel sistema non siano presenti link (né fisici né simbolici). Quanti pathname *relativi* sono associati a quel file? (motivate la vostra risposta).

Il file possiede un pathname per ciascuna cartella del sistema.

d) Nel sistema del punto a) si viene a sapere che tutti i file devono poter essere acceduti in modo diretto, e occupano sempre più di un blocco. Quale modalità verrebbe adottata per tenere traccia dei file? Quale svantaggio avrebbe comunque questa modalità, nel caso di file che occupano comunque pochi blocchi?

L'allocazione indicizzata. Per file piccoli, quasi tutto il blocco indice viene sprecato.