

II Prova in itinere – 17 Dicembre 2018  
(teoria)

Università di Salerno

1. **Codice comportamentale.** Durante questo esame si deve lavorare da soli. Non si può consultare materiale di nessun tipo. Non si può chiedere o dare aiuto ad altri studenti.
2. **Istruzioni.** Rispondere alle domande. Per la brutta usare i fogli posti alla fine del plico (NON si possono usare fogli aggiuntivi); le risposte verranno corrette solo se inserite nello spazio ad esse riservate oppure viene indicata con chiarezza la posizione alternativa.  
Per essere accettata per la correzione la risposta deve essere ordinata e di facile lettura.  
TUTTE le risposte vanno GIUSTIFICATE. Ciascuna risposta non giustificata vale ZERO.

Nome e Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_

Firma \_\_\_\_\_

Spazio riservato alla correzione: non scrivere in questa tabella.

1	2	Tot	bonus
/35	/15	/50	/10

1. *35 punti*

In un sistema con paginazione, le pagine sono grandi 512 byte, la memoria é costituita da 8 frame e la tabella delle pagine di un processo può avere al più 16 entry.

a) individuare la struttura dell'indirizzo fisico;

b) individuare la struttura dell'indirizzo logico.

- c) Si assuma che il processo  $P$  viene mandato in esecuzione all'istante 100 e che
- al processo vengono assegnati ai primi 3 frame della memoria principale e che essi sono vuoti quando il processo viene mandato in esecuzione (all'istante 100), e che
  - ogni accesso alla pagina 0 é in scrittura, e che
  - il processo fa riferimento nell'ordine (a partire dall'istante 100) alle seguenti pagine:

3, 0, 1, 1, 0, 5, 4, 2, 5, 0, 9

Mostrare istante per istante quale sono le pagine contenute nei 3 frame assegnati a  $P$  (riempiendo la tabella sottostante), nel caso in cui

- c1) l'algoritmo di sostituzione delle pagine **FIFO**

# frame	100	101	102	103	104	104	105	107	108	109	110

- Quale é il numero di page fault?
- Quale é il numero di accessi al disco?
- Se nell'istante 111 il processo  $P$  fa riferimento al "primo byte" della pagina 0, mostrare l'indirizzo logico generato dalla CPU ed il corrispondente indirizzo fisico.

istante	indirizzo logico	indirizzo fisico
111		

- c2) l'algoritmo di sostituzione delle pagine **LRU**

# frame	100	101	102	103	104	104	105	107	108	109	110

- Quale é il numero di page fault?
- Quale é il numero di accessi al disco?
- Se nell'istante 111 il processo  $P$  fa riferimento al "primo byte" della pagina 0, mostrare l'indirizzo logico generato dalla CPU ed il corrispondente indirizzo fisico.

istante	indirizzo logico	indirizzo fisico
111		

d) Siano  $P$  e  $Q$  i due unici processi presenti nel sistema all'istante 100. Nelle tabelle sottostanti sono riportati il contenuto delle tabelle delle pagine dei processi  $P$  e  $Q$  all'istante 100, dove è evidenziato per ciascuna pagina anche

- il valore del bit di validità (V) e del bit di modifica (M),
- l'istante di caricamento,
- l'istante dell'ultimo riferimento.

**PT processo  $P$** 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0		i			
1	4	v	0	40	99
2		i			
3	0	v	1	55	70
4	7	v	0	67	80
5		i			
6		i			
7		i			
8		i			
9		i			
10	2	v	0	80	82
11		i			
12		i			
13		i			
14		i			
15	3	v	1	73	73

**PT processo  $Q$** 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0		i			
1	6	v	0	50	98
2		i			
3		i			
4		i			
5		i			
6		i			
7	1	v	1	72	75
8	5	v	1	79	81
9		i			
10		i			
11		i			
12		i			
13		i			
14		i			
15		i			

d1) Si riempia la tabella sottostante che rappresenta lo stato della memoria principale all'istante 100. Per ciascun frame individuare la coppia (processo, pagina); cioè, se la pagina 4 del processo  $P$  è memorizzata nel frame 7 allora la coppia  $(P,4)$  deve essere scritta in corrispondenza del frame 7.

# frame	(processo, pagina)
0	
1	
2	
3	
4	
5	
6	
7	

d2) Si considerino i due seguenti accessi a memoria eseguiti in sequenza:  
 istante 101 — il processo  $P$  scrive nella pagina 4;  
 istante 102 — il processo  $Q$  legge dalla pagina 9.

d2.1) Assumendo che il sistema utilizzi un algoritmo di sostituzione **FIFO (globale)**, giustificando la risposta riportare nelle tabelle sottostanti le sole modifiche apportate alle tabelle dei processi  $P$  e  $Q$  ed a quella che mostra l'allocazione dei frame alla fine dell'istante 102.

PT processo  $P$ 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

PT processo  $Q$ 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

# frame	(processo, pagina)
0	
1	
2	
3	
4	
5	
6	
7	

d2.2) Assumendo che il sistema utilizzi un algoritmo di sostituzione **LRU (globale)**, giustificando la risposta riportare nelle tabelle sottostanti le sole modifiche apportate alle tabelle dei processi  $P$  e  $Q$  ed a quella che mostra l'allocazione dei frame alla fine dell'istante 102.

**PT processo  $P$** 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

**PT processo  $Q$** 

# page	# frame	V	M	ist. caric.	ultimo ist. rifer.
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

# frame	(processo, pagina)
0	
1	
2	
3	
4	
5	
6	
7	

## 2. 15 punti

Un fast food serve porzioni di patatine. Un garzone frigge le patatine e prepara vassoi con 20 porzioni. La padrona serve i clienti e quando arriva un cliente che chiede una porzione di patatine lei ne prende una dal vassoio e la dá al cliente. Se però il vassoio é vuoto, lei segnala la cosa al garzone che prepara un nuovo vassoio.

Si scrivano le procedure (in pseudocodice) che descrivono il comportamento dei tre processi **garzone**, **padrona** e **cliente**.

SUGGERIMENTO: Si utilizzino le seguenti variabili condivise:

```
semaphore servizio = 1; arrivo = 0; prepara = 0; vassoio = 0;  
int porzioni = 20;
```

## 3. 10 punti (bonus)

Si considerino i seguenti processi eseguiti correntemente sulla stessa CPU, con variabili condivise:

```
semaphore S, T;
int x=0;
```

Processo $P_1$	Processo $P_2$
while (1)	while (1)
{	{
wait(S)	wait(T)
x=x+1	write(x)
signal(T)	signal(S)
}	}

Giustificando le risposte, rispondere alle seguenti domande:

- 1) Come inizializzereste i semafori S e T in modo che l'output sia

0 1 2 3 4 ...

- 2) Come inizializzereste i semafori S e T in modo che l'output sia

1 2 3 4 ...

- 3) Quale inizializzazione dei semafori S e T causerebbe race condition?



FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA