

Prendete il vostro numero di matricola. Eliminate tutti gli "0", e chiamate N il numero risultante, e m la sua lunghezza. Esempio se il numero di matricola e' 0000800470, allora $N = 847$ e $m = 3$

Nel seguito N_i e' la cifra di N in posizione $i \bmod m$. Cioe': si contano le cifre di N da sinistra verso destra, la prima cifra e' N_0 , poi N_1 e cosi' di seguito, modulo la lunghezza di N . Nell'esempio sopra:

- $N_0 = 8$
- $N_1 = 4$
- $N_2 = 7$
- $N_3 = 8$
- $N_4 = 4$

Inoltre allo stesso modo N_2N_3 e' il numero 78 (e **non la moltiplicazione** 7×8).

Negli esercizi sotto: non eseguite calcoli o ragionamenti astratti in termini di questi N_i , ma sostituite subito ai vari N_i il valore concreto dato dal vostro numero matricola.

Ad inizio foglio indicate chiaramente il vostro numeri di matricola e quali sono nel vostro caso i numeri N_0, N_1, N_2, N_3 . Inoltre all'inizio di ogni esercizio in cui intervengono tali numeri, indicatene il valore

Motivare le risposte date. Non e' necessario dare risposte lunghe, ma e' importante rispondere in modo motivato ed esauriente alle domande poste (in altre parole, molto meglio una frase in piu' che una in meno).

Ricordare che lo scan finale deve contenere il libretto universitario di riconoscimento; e che il compito va inviato in pdf e in un unico file, come allegato

E' necessario svolgere gli esercizi 1 , 2, 3

Esercizio 1 Considerate i seguenti processi:

P1	P2	P3	P4
print(N_1)	print(N_2)	print(N_2)	print(N_3)

Agite, se necessario, sul codice, inserendo opportune operazioni su semaforo, in modo che l'output di questa esecuzione concorrente sia una stringa in cui i numeri stampati appaiano in ordine *non decrescente*.

In caso di possibilita' di non-determinismo (esempio, quello prodotto dai processi P2 e P3, che stampano lo stesso numero): meglio se la vostra soluzione riflette tale non-determinismo (esempio, nel caso di P2 e P3, lascia aperta la possibilita' che esegua per primo uno qualunque dei 2 processi).

Risposta(Sketch) Assumo $N_1 = 4, N_2 = 6, N_3 = 1$. Quindi occorre stampare la stringa 1466

Semafori S, T inizializzati a 0

P1	P2	P3	P4
S.P()	T.P()	T.P()	
print 4	print 6	print 6	print 1
T.V()			S.V()
T.V()			

Esempio di altra situazione interessante:

P1	P2	P3	P4
S.P()			
S.P()	T.P()	T.P()	
print 4	print 3	print 3	print 1
	S.V()	S.V()	T.V()
			T.V()

Esercizio 2 Sia: N il max tra N_2 e 5; ed M il max tra N_1 e 6.

Quattro processi arrivano al tempo indicato e consumano la quantita' di CPU indicata nella tabella sottostante:

Processo	T. di arrivo	Burst time
A	0	$N_2 + 7$
B	4	$N_3 + 4$
C	7	M
D	10	N

1. Riportate il diagramma di Gantt che mostra cio' che succedera' in esecuzione, adottando uno scheduling di tipo Round Robin con quanto di tempo 3.
2. Lo stesso, in questo caso assumendo Shortest Job First di tipo non-premptive.
3. Calcolare il waiting time di B nel 2o caso.

Risposta(Sketch) Assumiamo $N_1 = 5, N_2 = 4, N_3 = 6$ quindi $N = 5, M = 6$.

Processo	T. di arrivo	Burst time
A	0	11
B	4	10
C	7	6
D	10	5

[0]A[3]A[6]B[9]A[12]C[15]B[18]D[21]A[23]C[26]B[29]D[31]B[32]
Evoluzione processi nella ready-queue :

0 A
3 A
6 BA
9 ACB
12 CBDA
15 BDAC
18 DACB
21 ACBD
23 CBD
26 BD
29 DB
31 B
32

SJF:

[0]A[11] D[5] C[6] B [32]

WT (B) = 18 (22-4)

Esercizio 3 Spiegare cosa e' il Process Control Block, cosa contiene (almeno 2 campi), e a cosa serve.

Risposta(Sketch) Vedere note corso.

Esercizio 4 []

1. Nel caso di politica di Optimal Replacement, mostrare i contenuti dei frame della memoria dopo ogni referenza a pagina, data la reference string

1,6,1,4,1,2,5,7,3

e 4 frame a disposizione con contenuto iniziale: 4 6 7 3. Indicare chiaramente i page fault. Quanti ne occorrono?

2. Per il punto sopra, calcolare il tempo medio di accesso assumendo che: il tempo di swap e' 20 millisecondi; il tempo di accesso alla memoria e' 10 microsecondi; in media 500 accessi sono effettuati ad una pagina una volta che questa e' chiamata (cioe' ogni volta che compare nella reference string).

Risposta(Sketch)

1. Ci sono 4 page fault. Ci sono varie evoluzioni possibili, tra cui questa:

4 2
 6 5
 7
 3 1 3

Si poteva finire anche con, esempio

2 4 5
 6 oppure 6 2
 7 3 3
 3 5 1

2. $((4 \cdot 20 \cdot 1000) + 9 \cdot 10 \cdot 500) / 9 \cdot 500$

(dove 4 e' dato dal numero di page fault, e 9 sono le pagine utilizzate, ognuna 500 volte).

Esercizio 5 Si consideri un sistema in cui:

- Un indirizzo fisico del sistema e' scritto su $(20 + N_2)$ bit, e la RAM e' suddivisa in 2^{N_3+10} frame.
- in una tabella delle pagine di un processo, l'indice piu' grande usabile nella tabella delle pagine di quel processo puo' essere

$$\underbrace{1 \dots 1}_{N_4+10} \text{ (in binario).}$$

1. Come e' fatto l'indirizzo fisico ?
2. Quanto e' grande una pagina?
3. Quanto e' grande lo spazio di indirizzamento logico del sistema (cioe' il processo piu' grande che il sistema ammette) ?
4. Il sistema deve usare una paginazione a piu' livelli? (motivate numericamente la vostra risposta).

Risposta(Sketch) Risposta, nel caso $N_4 = 5, N_3 = 3, N_2 = 5$.

1. 25 bit di IF; 13 bit di n. frame
Quindi 12 bit di offset

$$\begin{array}{cc} \text{[N frame]} & \text{[offset]} \\ 13 & 12 \end{array}$$

2. un numero di frame e' scritto su 13 bit, e la dimensione di un frame, e quindi di una pagina, e' di 2^{12} byte ($25 - 13 = 12$).
3. Poiche' il numero piu' grande di una pagina e' 111111111111111, ci possono essere al massimo 2^{15} pagine, e lo spazio di indirizzamento logico e' di $2^{15} \times 2^{12}$ byte , cioe' 2^{27} byte
4. Si. La tabella delle pagine piu' grande del sistema ha 2^{15} entry, e ogni entry contiene il numero di un frame, per cui sono necessari almeno due byte per ogni entry. La dimensione di quella tabella sara' quindi di $2^{15} \times 2 = 2^{16}$ byte ossia maggiore della dimensione di un frame.

Esercizio 6 Sia $N = N_1$, se N_1 diverso da 1; altrimenti N e' il max tra 3 e N_2 .
Considerate i 2 seguenti threads

T_1 = while true do print A

T_2 = while true do print B

Questi 2 thread vengono lanciati in parallelo. Agite, se necessario, sul codice in modo che l'output di questa esecuzione concorrente sia sempre la sequenza

$$\underbrace{A \dots A}_N B \underbrace{A \dots A}_N B \dots$$

Le uniche modifiche possibili sono aggiunta di operazioni su semafori.

Risposta(Sketch) Supponiamo che N sia 2 (per altri numeri, la soluzione e' simile)

$$\begin{aligned}T_1 &= \textit{while true do } T.P(); \textit{ printA}; S.V() \\T_2 &= \textit{while true do } T.V(); T.V(); S.P(); S.P(); \textit{ printB};\end{aligned}$$

con semafori = 0

Esercizio 7 1. Nella cifratura a chiave simmetrica, perche' una permutazione sull'alfabeto non e' considerata una buona chiave?

2. Siano:

- $N = N_2$, se N_2 diverso da 7 e diverso da 1; $N = 6$ altrimenti;
- $M = N_1$, se N_1 diverso da 7 e diverso da 1; $M = 11$ altrimenti

Usare le proprieta' dell'aritmetica modulo per calcolare in modo semplice

$$(N^{11} \times M^{N_3+5}) \bmod 7$$

Risposta(Sketch)

1. note corso

2. soluzione per $N_2 = 6, M = 11, N_3 = 4$.

$$(6^{11} \times 11^9) \bmod 7$$

$$(6^{11} \times 4^9) \bmod 7$$

poiche' $11 \bmod 7 = 4$.

Inoltre abbiamo $6^2 \bmod 7 = 36 \bmod 7 = 1$

Quindi anche $6^4 \bmod 7 = 6^8 \bmod 7 = 1$.

Siccome $11 = 8 + 4 + 1$, $6^{11} \bmod 7 = 6 \bmod 7 = 6$.

Similmente $4^2 \bmod 7 = 16 \bmod 7 = 2$

Quindi

$$4^4 \bmod 7 = 2^2 \bmod 7 = 4$$

$$4^8 \bmod 7 = 4^2 \bmod 7 = 2$$

Quindi $4^9 \bmod 7 = (4 \times 2) \bmod 7 = 1$

Quindi $(6^{11} \times 4^9) \bmod 7 = 6$