

Ricordarsi di mettere il proprio nome, cognome, e numero di matricola in tutti i fogli. Motivare sempre le risposte date. Non e' necessario dare risposte molto 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).

Per avere la sufficienza, e' **necessario**** svolgere tutti i primi 4 esercizi.**

Non sono ammesse macchinette calcolatrici o altre macchine elettroniche; non e' consentito uso di appunti o libri.

Malacopia: consegnare, se necessario **solo** gli esercizi che devono essere corretti (non riportati in bella copia); barrare quindi gli altri

Esercizio 1 [] Considerate i 3 processi sotto

```
print(A);          print(B);          print(C);
print(D);          print(E);
print(F);          print(G);
```

Agite, se necessario, sul codice, inserendo opportune operazioni su semaforo, in modo che l'output di questa esecuzione concorrente sia sempre la stringa ADBCFEF. Indicare bene i semafori utilizzati e il loro valore di inizializzazione.

Risposta(Sketch)

```
print(A);          T.P()          U.P()
print(B);          print(B);      print(C);
print(D);          U.V(); W.P()  Z.V()
T.V(); Z.P()      print(E);
print(F);          print(G);
W.V()
```

con inizialmente tutti i semafori a 0

Esercizio 2 1. Qual'e' la differenza tra "modalita' utente" e "modalita' monitor" (nell'esecuzione di codice su un processore)?

2. E' vero che una interrupt viene gestita in modalita' utente? Perche'?

Risposta(Sketch) appunti corso

Esercizio 3 Descrivere il metodo di allocazione di file su disco detto 'indexed'. Come funziona? che vantaggi e svantaggi presenta?

Risposta(Sketch) appunti corso

Esercizio 4 Perche' puo' essere vantaggioso usare thread al posto di processi ?

Risposta(Sketch) appunti corso

Esercizio 5 Si consideri la seguente serie di riferimenti a pagine di memoria:

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

Si considerino le seguenti politiche di rimpiazzo:

- LRU
- Optimal

Quanti page fault avvengono considerando una RAM con solo 4 page frame ed inizialmente vuota? Si mostri l'evoluzione dei frame.

Risposta(Sketch) Assumendo di iniziare con 1,2,3,4; si finisce con 6,2,3,1 (LRU) e con 1,2,3,6 (optimal), e con, rispettivamente, 10 e 8 page fault

Esercizio 6 Un sistema ha una memoria RAM formata da 512 frames di 256 byte ciascuno. Lo spazio di indirizzamento logico e' grande 8 volte quello fisico.

1. Qual e' l'indirizzo fisico piu' grande ammesso dal sistema?
2. Qual e' l'indirizzo logico piu' grande ammesso dal sistema?
3. In quante pagine e' suddiviso lo spazio di indirizzamento logico del sistema?
4. Se la tabella delle pagine e' mantenuta esclusivamente in RAM, di quanto si degradano approssimativamente le prestazioni del sistema rispetto ad un sistema equivalente ma senza paginazione della memoria?
5. In pratica, la tabella delle pagine, in quale posto (o quali posti) e' mantenuta)? Perche' questo limita l'eccessiva perdita di prestazioni del sistema?
6. Cosa e' la Inverted Page Table? (cioe', qual'e' l'idea del suo funzionamento)?

Risposta(Sketch)

1. La RAM e' formata da $2^9 * 2^8 = 2^{17}$ celle, il cui indirizzo va da 0 a $2^{17} - 1$
2. Lo spazio logico e' pari a $2^3 * 2^{17} = 2^{20}$
3. Lo spazio di indirizzamento logico e' suddiviso in $2^{20}/2^8 = 2^{12}$ (circa 2000) pagine.
4. Il tempo medio di esecuzione dei programmi raddoppia, poiche' il tempo di accesso ad ogni cella di memoria (dovendo passare attraverso la tabella delle pagine) raddoppia
5. Si usa una memoria associativa di supporto (TLB). Vedere appunti di corso.
6. Vedere appunti di corso.

Esercizio 7 Si consideri il problema dei lettori e scrittori visto a lezione, dove i codici del generico scrittore e del generico lettore sono riportati qui di seguito. Inserite le operazioni di decremento/incremento (dette anche wait e signal, oppure P e V) su semaforo mancanti necessarie per il corretto funzionamento del sistema, indicando anche il semaforo mancante. Ricordarsi dei valori di inizializzazione. Come per tutte le risposte, motivate brevemente anche le aggiunte fatte.

```
semafori e variabili condivise necessarie
semaphore write = 1;
int numlettori = 0;

scrittore {
```

```

...Esegui la scrittura del file ...

}

lettore {

numlettori++;
if numlettori == 1 ;
... leggi il file ...
numlettori--;
if numlettori == 0 ;
}

```

Risposta(Sketch)

semafori e variabili condivise necessarie

```

semaphore mutex = 1;
semaphore write = 1;
int numlettori = 0;

scrittore {
wait(scrivi);
..Esegui la scrittura del file ...
signal(scrivi) }

lettore {
wait(mutex);
numlettori++;
if numlettori == 1 wait(scrivi);
signal(mutex);
... leggi il file ...
wait(mutex);
numlettori--;
if numlettori == 0 signal(scrivi);
signal(mutex); }

```

Esercizio 8 La soluzione del problema dei lettori e scrittori nell'esercizio sopra garantisce l'assenza di starvation?

Risposta(Sketch) No, infatti un qualsiasi processo scrittore potrebbe dover attendere all'infinito senza riuscire ad entrare in sezione critica. Al contrario i processi lettori sono liberi da starvation (in altre parole, non rantita l'attesa limitata)

Esercizio 9 Descrivere una soluzione al problema delle firme digitali usando la crittografia.