**Compito di Sistemi Operativi, Prof. Mancini**

19 giugno 2001

1. (punti: -1,4)
   L'algoritmo di Peterson:
   (a) realizza la mutua esclusione sulle sezioni critiche di processi concorrenti;
   (b) non realizza la mutua esclusione sulle sezioni critiche di processi concorrenti;
   (c) evita l'attesa indefinita e presuppone l'utilizzo di istruzioni speciali;
   (d) evita l'attesa indefinita e non presuppone l'utilizzo di istruzioni speciali;
   (e) non evita l'attesa indefinita e presuppone l'utilizzo di istruzioni speciali;
   (f) non evita l'attesa indefinita e non presuppone l'utilizzo di istruzioni speciali;
   (g) non da luogo ad interferene sebbene non venga usato alcuno strumento per la mutua esclusione, questo perchè la muta esclusione è solo una condizione sufficiente per la corretta realizzazione delle interazioni attraverso dati comuni;
   (h) non da luogo ad interferene sebbene non venga usato alcuno strumento per la mutua esclusione, questo perchè la muta esclusione è solo una condizione necessaria per la corretta realizzazione delle interazioni attraverso dati comuni;
   (i) nessuna delle affermazioni precedenti è corretta.

2. (punti: -1,4)
   L'organizzazione della memoria centrale con partizioni variabili:
   (a) alloca ad un programma uno spazio fisico di memoria centrale di ampiezza minore di quella dello spazio logico del programma;
   (b) alloca ad un programma uno spazio fisico di memoria centrale di ampiezza uguale a quella dello spazio logico del programma;
   (c) elimina la frammentazione interna e rimuove il vincolo della contiguità dello spazio fisico in memoria centrale;
   (d) elimina la frammentazione interna ma mantiene il vincolo della contiguità dello spazio fisico in memoria centrale;
   (e) introduce una certa frammentazione interna, che è tanto meno rilevante quanto più la lunghezza media dei programmi è grande;
   (f) elimina il problema della frammentazione esterna;
   (g) introduce una certa frammentazione esterna;
   (h) nessuna delle affermazioni precedenti è corretta.

3. (punti: -1,4)
   Nella gestione dinamica della memoria, l'algoritmo di sostituzione delle pagine:
   (a) permette di ricavare i dati significativi per la valutazione della gestione del memoria per mezzo di un modello probabilistico del sistema;
   (b) permette di rilevare i dati significativi per la valutazione della gestione del memoria durante il normale funzionamento del sistema operativo;
   (c) stima la probabilità di riferimento futuro sulla base dei riferimenti avvenuti nel passato recente;
   (d) può essere globale oppure locale;

(e) include l'algoritmo LRU che può essere definito in termini di *distanza passata*;

(f) include l'algoritmo LRU che può essere definito in termini di *distanza futura*;

(g) include l'algortmo Second Chance che organizza le pagine accessibili in un vettore circolare, e non associa ad ognuna di queste pagine un indicatore booleano di pagina riferita;

(h) nessuna delle affermazioni precedenti è corretta.

4. (punti: -1,4)
Un elemento E della struttura dati "tabella dei collegamenti" del sistema di archiviazione di Unix è associato ad un archivio F.

(a) L'elemento E contiene fra l'altro il puntatore alla posizione corrente nell'archivio F,

(b) L'elemento E contiene fra l'altro il modo con il quale l'archivio F è stato aperto;

(c) L'elemento E non contiene il puntatore alla posizione corrente nell'archivio F, e non contiene il modo con il quale l'archivio F è stato aperto;

(d) L'elemento E contiene fra l'altro l'indice dell'elemento della tabella degli i-node in memoria centrale che mantiene l'i-node dell'archivio F;

(e) L'elemento E contiene fra l'altro un contatore che registra la cardinalità dell'insieme di processi al quale l'elemento E è associato;

(f) L'elemento E non contiene l'indice dell'elemento della tabella degli i-node in memoria centrale che mantiene l'i-node dell'archivio F e non contiene un contatore che registra la cardinalità dell'insieme di processi al quale l'elemento E è associato;

(g) L'elemento E non contiene il puntatore alla posizione corrente nell'archivio F, ma contiene l'indice dell'elemento della tabella degli i-node in memoria centrale che mantiene l'i-node dell'archivio F ed un contatore che registra la cardinalità dell'insieme di processi al quale l'elemento E è associato;

(h) nessuna delle affermazioni precedenti è corretta.

5. (punti: -1,4)
Ad ogni processo del sistema operativo Unix viene associata una quadrupla (*proprietario reale, gruppo reale, proprietario effettivo, gruppo effettivo*):

(a) il proprietario reale e il gruppo reale contengono informazioni (UID e GID) utilizzate internamente dal sistema per determinare in modo univoco l'utente, mentre il proprietario effettivo e il gruppo effettivo vengono utilizzati per realizzare la migrazione nei domini di protezione di altri utenti;

(b) il proprietario reale e il gruppo reale contengono informazioni (ad esempio, username) relative all'utente reale, mentre il proprietario effettivo e il gruppo effettivo contengono informazioni (UID e GID) utilizzate internamente dal sistema per determinare in modo univoco l'utente;

(c) il proprietario reale e il gruppo reale contengono informazioni (UID e GID) utilizzate internamente dal sistema per determinare in modo univoco l'utente, mentre il proprietario effettivo e il gruppo effettivo contengono informazioni (ad esempio, username) relative all'utente reale;

(d) l'identificazione degli utenti si propaga ai processi, nell'istante di creazione ogni processo è attribuito ad un utente proprietario assegnando opportunamente i valori alla quadrupla che non può mai essere modificata durante l'esecuzione del processo;

(e) l'identificazione degli utenti si propaga ai processi, nell'istante di creazione ogni processo è attribuito ad un utente proprietario assegnando opportunamente i valori alla quadrupla che può essere modificata durante l'esecuzione del processo;

(f) i meccanismi di protezione di Unix non consentono la migrazione dei processi in domini di protezione diversi da quelli dei loro proprietari;

(g) i meccanismi di protezione degli archivi di Unix sono basati sul presupposto della sicura identificazione degli utenti;

(h) si assuma che un processo invochi la chiamata di sistema **exec** per eseguire il programma contenuto in un archivio $F_i$; **exec** assegna al proprietario reale, e al gruppo reale i valori corrispondenti al proprietario dell'archivio $F_i$ se e solo se è vera la condizione SETUID contenuta nell' i-node di $F_i$;

(i) si assuma che un processo invochi la chiamata di sistema **exec** per eseguire il programma contenuto in un archivio $F_i$; **exec** assegna al proprietario effettivo, e al gruppo effettivo i valori corrispondenti al proprietario dell'archivio $F_i$ se e solo se è vera la condizione SETUID contenuta nell' i-node di $F_i$;

(j) nessuna delle affermazioni precedenti è corretta.

6. (punti: 6)
Usando un massimo di 100 parole, definire la nozione di semaforo e discutere la differenza tra il modello booleano ed il modello produttore-consumatore.

7. (punti: 6)
Illustrare in al più 70 parole i principali inconvenienti della gestione della memoria centrale con partizioni fisse.

8. (punti: 6)
Considerare un insieme di cinque processi $P_1$, $P_2$, $P_3$, $P_4$, $P_5$ con i seguenti tempi di arrivo e tempi di esecuzione in millisecondi:

| Processo | Tempo di arrivo | Tempo di esecuzione |
|---|---|---|
| $P_1$ | 0 | 7 |
| $P_2$ | 15 | 8 |
| $P_3$ | 3 | 20 |
| $P_4$ | 11 | 14 |
| $P_5$ | 23 | 13 |

Assegnare questo insieme di processi ad un processore in base alla politica Round Robin considerando un quanto di tempo di 6 millisecondi.

Calcolare il valor medio del tempo di attesa ed il valor medio del tempo di turnaround dei processi.

9. Considerare la seguente stringa di riferimenti alla memoria di un processo in un sistema con memoria virtuale
S = 1 12 3 2 0 10 2 12 9 10 5 2 1 3 9 5 0 9 0 2

(a) (punti: 3)
Illustrare il comportamento dell'algoritmo LRU di sostituzione delle pagine per una memoria fisica di 5 blocchi. Calcolare il numero di page fault che si verificano.

(b) (punti: 3)
Illustrare il comportamento dell'algoritmo Second Chance di sostituzione delle pagine per una memoria fisica di 5 blocchi. Calcolare il numero di page fault che si verificano.

10. (punti: 16)
Scrivere una implementazione MyHandShake della interfaccia HandShake che consenta ai due Gentleman del seguente programma di sincronizzarsi con uno "handshake": in nessuna esecuzione del programma HowDoYouDo la parola "dopo" non puo' mai essere scritta prima della parola "prima".

```
interface HandShake {
    public void shakeHands ();
}

class Gentleman extends Thread {

    static HandShake H = new MyHandShake();

    public void run () {
        System.out.println("prima");
        H.shakeHands();
        System.out.println("dopo");
    }
}

public class HowDoYouDo {

    public static void main (String [] args) {
        (new Gentleman()).start();
        (new Gentleman()).start();
    }
}
```

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

**Risposte Compito di Sistemi Operativi, 19/6/2001**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. (punti: -1,4) | **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** |
| 2. (punti: -1,4) | **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** |
| 3. (punti: -1,4) | **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** |
| 4. (punti: -1,4) | **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** |
| 5. (punti: -1,4) | **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** |

6. (punti: 6)

7. (punti: 6)

8. (punti: 6)

9. (a) (punti: 3)

   (b) (punti: 3)