



Corso di Laurea in Ingegneria Informatica -- **Sistemi Operativi** App. 13 Luglio 2012

1. Che cosa è l'anomalia di Belady? [max 4 pt]
2. Si descriva il file system UFS (Unix File System) e la struttura dell'*i-node*. [max 6 pt]
3. **a.** Descrivere l'algoritmo *Hard Real-Time a frequenza monotona*. [max 5 pt]  
**b.** Si considerino i processi (t è il tempo di picco, p il periodo e d la scadenza):

Processo	t	p=d	% uso CPU
$P_1$	30	100	0.30
$P_2$	10	40	0.25
$P_3$	20	50	0.4

[max 5 pt]

e si esegua l'algoritmo di cui al punto a., effettuando il controllo di ammissione e disegnando il diagramma di Gantt fino a 210 ms. L'esempio dato è schedulabile?

4. **Quesito per gli studenti del C.I. di Reti di calcolatori e Sistemi operativi:** [max 10 pt]  
Nell'ambito dei metodi di sincronizzazione dei processi, si consideri il problema dei lettori/scrittori con priorità agli scrittori: nessun scrittore deve essere tenuto in attesa, a meno che i lettori hanno già ottenuto l'accesso alla base di dati condivisa. Si definisca, con un pseudolinguaggio di programmazione, la classe Database contenente i metodi per acquisire i lock di lettura e di scrittura facendo uso dei semafori.
5. **Quesito riservato agli studenti di Informatica II (21013+23014):** Si consideri un sito web dedicato ai collezionisti di figurine dell'album *Campionato europeo di calcio 2012*. L'album è composto da 100 diverse figurine, ognuna individuata univocamente da un intero. Alcune sono classificate come figurine rare, e le rimanenti come figurine normali a seconda della quantità  $i$  disponibile: se  $i < \text{maxrare}$ , si tratta di una figurina rara; se  $i \geq \text{maxrare}$ , si tratta di una figurina normale. Il sito offre un servizio che permette ad ogni utente collezionista (un thread!) di effettuare scambi di figurine. A questo scopo il sistema gestisce un deposito di figurine (la risorsa condivisa!), nel quale, per ogni diversa figurina vi può essere più di un esemplare. Il meccanismo di scambio, è regolamentato come segue:
  1. Si può scambiare solo una figurina alla volta;
  2. **Richiesta di scambio:** ogni utente U che desidera una figurina A può ottenerla, se a sua volta offre un'altra figurina B; in seguito a una richiesta di scambio, il sistema aggiunge la figurina B all'insieme delle figurine disponibili e successivamente verifica se esiste almeno una istanza di A disponibile: **a.** se A è disponibile, essa viene assegnata all'utente, che può così continuare la propria attività; **b.** se A non è disponibile, l'utente U viene messo in attesa.
  3. **Vincolo di precedenza:** le richieste di utenti che offrono figurine rare hanno la precedenza sulle richieste di utenti che offrono figurine normali;

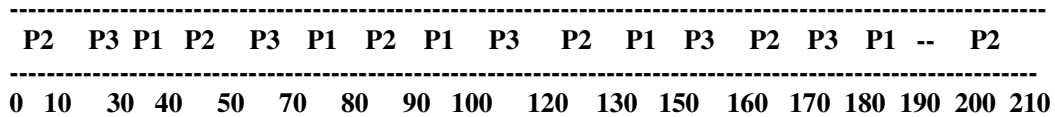
Si definisca la politica di scambio con i semafori, completando la classe Java Deposito. [max 10 pt]

```
public class Deposito{
    private final int N =100; //costante che esprime il numero totale di figurine
    private final int maxrare; //soglia per la proprietà dinamica "figurina rara"
    private int[] FIGURINE; //FIGURINE[i] è il numero di esemplari disponibili della figurina i
    ... <completare>
    public Deposito( ) { //Costruttore:
        FIGURINE=new int[N];
        maxrare=N/3; //nel testo sarebbe 30, ma in questo modo posso parametrizzare
        ... <completare>
    }
    //metodo invocato dai thread collezionisti per lo scambio delle figurine
    public void scambio(int off, int rich) throws InterruptedException { ... <completare> }
}
```

**SOLUZIONE:**

1. Vedi libro di testo/lucidi della lezione relativa.
2. Vedi libro di testo/lucidi della lezione relativa.
3. Il campione di processi è ammesso alla schedulazione perché la somma delle percentuali  $t/p$  è  $0,3+0,25+0,4=0,95$  ovvero  $95\% < 100\%$ . Inoltre, in base alla priorità statica (l'inverso del periodo), i processi vengono eseguiti nell'ordine: P2, P3, P1. P1 si ripresenta/scade ( $p=d$ ) agli istanti 100, 200, 300, ecc..; P2 si ripresenta/scade ( $p=d$ ) a 40, 80, 120, 160, 200, ecc..; e P3 si ripresenta/scade ( $p=d$ ) a 50, 100, 150, 200, ecc..

In base al diagramma di *Gantt* di seguito riportato, il campione di processi è schedulabile con tale algoritmo. Ogni processo viene, infatti, portato a termine (a volte anche al limite!) prima di ogni scadenza.



**P1** interrotto all'istante 40 da P2 che si ripresenta, restano 20 ms di P1  
**P1** nuovamente interrotto all'istante 80 da P2 che si ripresenta, restano 10 ms di P1  
**La prima esecuzione di P1 si completa appena in tempo a 100, e si ripresenta a 100**  
**P1** viene interrotto a 150 da P3 che si ripresenta, restano 10 ms di P1  
**P3** viene interrotto a 160 da P2 che si ripresenta, restano 10 ms di P3  
**La quarta esecuzione di P3 si completa a 180 (scadrebbe a 200)**  
**La seconda esecuzione di P1 si completa a 190, seguono 10 istanti di inattività e poi si ricomincia a 200 (dove si ripresentano tutti) con P2.**

4. Omessa.  
*Suggerimento:* occorre impiegare un semaforo binario *mutex* per la mutua esclusione ed un array di semafori *sem[N]* generalizzati (uno per per figurina).

## ESITO PROVA SCRITTA

BETTONI DANIEL	<b>7</b>	
CORNELLI FABIO	<b>22</b>	
DE ANGELIS DOMENICO	<b>15</b>	
ENRIQUEZ CALLE MAURICIO		<b>26</b>
GALIZZI FRANCESCO	<b>27</b>	
GELMI ALBERTO	<b>5</b>	
GRISSETTI GIANANDREA	<b>21</b>	
HAXHI NIKOLLAQ	<b>19</b>	
LUBRINA STEFANO	<b>26</b>	
MARONI GABRIELE	<b>26</b>	
MARTINELLI MATTEO	<b>26</b>	
OPRENI SIMONE	<b>24</b>	
PEDRANA ANDREA	<b>27</b>	
PEZZOLI MATTEO	<b>21</b>	
PEZZOTTA YURI RICCARDO		<b>16</b>
PREVITALI ANDREA	<b>25</b>	
RONCOLATO ANDREA	<b>18</b>	
ROVETTA NICOLA	<b>16</b>	
SCANDELLA MATTEO	<b>28</b>	
SURDU VASILE	<b>14</b>	
VILLA DOMINIQUE	<b>9</b>	
CASATI LUCA	<b>25</b>	
FERNANDES DAFONSECA ANDREA		<b>26</b>