

**Università degli studi di Verona**  
**Corso di Laurea in Informatica**

**Sistemi Operativi**  
**10 Febbraio 2014**

---

**Chi fa l'esame completo deve svolgere tutti gli esercizi (2 ore di tempo). Chi deve fare solo la seconda parte deve svolgere solo gli esercizi 4 e 5 (1 ora di tempo).**

---

1. Il gestore di un parcheggio vuole automatizzare il sistema di ingresso/uscita in modo da impedire l'accesso quando il parcheggio risulta pieno (i.e., la sbarra d'ingresso deve sollevarsi solo se ci sono posti disponibili) e da consentire l'uscita solo in caso di parcheggio non vuoto (i.e., la sbarra d'uscita deve rimanere abbassata in caso di parcheggio vuoto). Scrivere un monitor che permetta di gestire l'accesso al parcheggio.

Suggerimento: Implementare all'interno del monitor una procedura *enter()* e una procedura *exit()*.

**[7 punti]**

2. Si descriva in dettaglio il concetto di deadlock.

Quindi, si consideri un sistema con  $m$  risorse della stessa tipologia contese da  $n$  processi, e si supponga che ogni processo  $p_i$  richieda al massimo  $max_i$  risorse delle  $m$  totali. Assumendo che le risorse possano essere richieste e rilasciate dai processi una alla volta, sotto quali condizioni (ovvero per quali valori di  $m$ ,  $n$ ,  $max_i$ ) il sistema può essere considerato deadlock free?

Suggerimento: si deve trovare un'equazione che lega  $m$ ,  $n$  e  $max_i$  ( $i = 1, \dots, n$ ).

**[4+4 punti]**

3. Si consideri il seguente insieme di processi:

Processo	CPU burst	Tempo di arrivo
1	1.5	0.0
2	3.5	1.0
3	0.5	0.5
4	2.0	3.0
5	1.0	1.0

Si mostri il diagramma dell'esecuzione dei processi usando gli algoritmi di scheduling HRRN, e RR con quanto pari a

1. Si calcoli il tempo di risposta, attesa e turnaround per ogni processo.

**[6 punti]**

- 
4. Si descrivano le fasi necessarie per trasformare un programma in processo, si spieghi quindi il concetto di binding degli indirizzi e infine si presentino le differenti tipologie di collegamento (linking) e caricamento (loading). **[7 punti]**

5. Si descriva in dettaglio il concetto di memoria virtuale facendo esplicito riferimento a un contesto di allocazione dello spazio tramite paginazione.

Quindi, data la seguente reference string, 0 1 2 3 0 2 1 2 1 0 1 2 3, calcolare il numero di page fault che si verificano applicando gli algoritmi LRU, FIFO e ideale, nel caso di 3 frame. Mostrare il contenuto della memoria. **[2+3 punti]**

---