

Esercizio -1: essersi iscritti correttamente per svolgere questa prova.

Esercizio 0: Scrivere correttamente nome, cognome e numero di matricola prima di svolgere ogni altro esercizio seguente.

Esercizio 1: Si consideri un banchiere che gestisca due valute, e si consideri lo stato seguente:

Valuta 1:				Valuta 2:		
i	C _i	P _i	N _i	C _i	P _i	N _i
IC=54; COH 4				IC=54; COH=6		
1	20	10	10	28	14	14
2	14	10	4	14	8	6
3	10	8	2	24	16	8
4	22	12	10	8	6	2
5	28	10	18	20	4	16

- A) Dimostrare che tale stato è safe per l'algoritmo del banchiere multivaluta.
- B) Mostrare un esempio di richiesta (della forma "il processo x richiede y unità della valuta z ") tale per cui (i) la richiesta sia soddisfacibile, dato il saldo di cassa attuale e (ii) la richiesta debba essere rifiutata dall'algoritmo del banchiere multivaluta, in quanto porterebbe ad uno stato unsafe. Mostrare lo stato risultante ed evidenziare il motivo del problema.

Esercizio 2: Si consideri l'algoritmo Rate Monotonic per processi real-time periodici. Si mostri un esempio di almeno quattro processi che NON SIANO schedulabili secondo la condizione di Liu and Layland, ma che SIANO effettivamente schedulabili dall'algoritmo. A tal scopo, mostrare il grafo di Gantt relativo allo scheduling di tale algoritmo. Per evitare casi banali, ogni processo deve avere un rapporto C/T (costo computazionale/periodicità) di almeno 0.14. Si ricorda che $4(2^{1/4} - 1)$ è circa uguale a 0.743.

Esercizio 3: Si consideri una memoria virtuale composta da 5 pagine, che devono essere allocate su una memoria fisica composta da 4 frame. Date queste condizioni, si mostrino tre casi distinti tali per cui: (A) l'algoritmo dell'orologio e MIN generino entrambi 7 page fault (o dimostrare che tale caso non esiste) (B) l'algoritmo dell'orologio abbia un numero di page fault maggiore del doppio di quelli generati da MIN (o dimostrare che tale caso non esiste) (C) l'algoritmo dell'orologio abbia un numero di page fault minore della metà di quelli generati da MIN (o dimostrare che tale caso non esiste)

Esercizio 4: Sia dato uno scheduler preemptive di tipo "Shortest-Remaining-Time-First" (ovvero uno scheduler che approssima SJF) con valore $\alpha = 1/2$ e valore iniziale $\tau_0 = 0$ per tutti i processi. Sia data la storia esecutiva dei seguenti processi:

P1: 2ms CPU, 5 ms I/O un.2, 2ms CPU, 5 ms I/O un 1, 3 ms CPU
P2: 4ms CPU, 5 ms I/O un.1, 4ms CPU, 5 ms I/O un 2, 3 ms CPU
P3: 5ms CPU, 5 ms I/O un.1, 6ms CPU, 5 ms I/O un 2, 3 ms CPU
P4: 3ms CPU, 5 ms I/O un.2, 3ms CPU, 5 ms I/O un 1, 4 ms CPU

mostrare il diagramma di Gantt e calcolare il tempo totale di esecuzione.