

Cognome e nome: _____ Matricola: _____ Posto: _____

Università degli Studi di Padova - Corso di Laurea in Informatica

Regole dell'esame

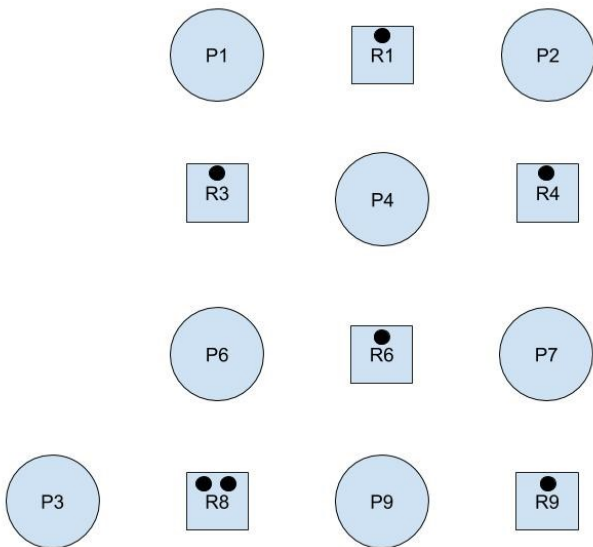
Il presente esame scritto deve essere svolto in forma individuale in un tempo massimo di 30 min dalla sua presentazione. Non è consentita la consultazione di libri o appunti in forma cartacea o elettronica, né l'uso di palmari e telefoni cellulari. Il candidato riporti generalità e matricola negli spazi indicati in alto e inserisca le proprie risposte interamente su questi fogli.

Quesito 1: 1 punto per risposta giusta, diminuzione di 0,5 punto per ogni sbaglio, 0 punti per risposta vuota

DOMANDA	Vero/Falso
Una <i>system call</i> da sempre luogo ad un <i>mode switch</i> tra modalità utente e modalità <i>kernel</i>	
Un <i>process switch</i> tra processi utente avviene sempre contestualmente a 2 <i>mode switch</i> (utente->kernel, kernel->utente)	
Un <i>interrupt</i> viene gestito in modalità <i>kernel</i>	
Ogni <i>interrupt</i> è associato ad un processo che ha richiesto una operazione di I/O	
L'inversione di priorità è una tecnica utilizzata per evitare la <i>starvation</i> dei processi a bassa priorità	
Un processo per lanciare un nuovo processo deve fare una <i>system call</i>	

Quesito 2:

Un sistema è composto dai seguenti processi e risorse (di cui solo la risorsa R8 ha molteplicità 2 anziché 1)



La situazione corrente del sistema è la seguente:

- P1 occupa R3, richiede R1
- P2 occupa R1, richiede R4
- P3 occupa una istanza di R8
- P4 richiede R1
- P6 occupa R6, richiede R3 e R8
- P7 occupa R4, richiede R9
- P9 occupa una istanza di R8 e R9, richiede R6

(Attenzione: questi sopra rappresentano una istantanea della situazione corrente e NON una sequenza cronologica. Dunque se leggete che P1 richiede R1, dovete lasciare la risorsa come NON assegnata).

Si determini, utilizzando il grafo di allocazione delle risorse, se il sistema sia in stallo (deadlock) e, in caso affermativo, quali siano i processi e le risorse coinvolti.

Cognome e nome: _____ Matricola: _____ Posto: _____

Quesito 3

Supponiamo di avere 2 processi che condividono una variabile x e che i loro pseudo-codici siano i seguenti (i numeri a sinistra delle istruzioni non fanno parte del codice, servono solo a identificare le istruzioni nei commenti di chi risolve l'esercizio):

P1:	P2:
1: P (SemA)	8: P (SemB)
2: P (Mutex)	9: P (Mutex)
3: if (x<0) then print(x)	10: x=x-2
4: V (Mutex)	11: V (Mutex)
5: V (SemC)	12: V (SemA)
6: P (SemA)	13: P (SemC)
7: print(x)	14: P (Mutex)
	15: x=x-1
	16: V (Mutex)
	17: print(x)

I due processi P1 e P2 tentano di eseguire in modo concorrente tra loro. Si assuma che il valore iniziale di x sia 1 e che i semafori abbiano i seguenti valori iniziali: SemA = 0, SemB = 1, SemC = 0, Mutex = 1.

[A] Determinare se e quali istruzioni `print(x)` saranno mai eseguite (indicare il numero alla sinistra dell'istruzione corrispondente) e in caso positivo dichiararne l'output.

[B] Si elenchi un possibile ordine di esecuzione delle istruzioni appartenenti ai due processi (Es. "1, 2, 3, 8, 9, 10, 11, 4, 5").

Quesito 4:

Un sistema ha 4 processi (A, B, C, D) e 5 risorse (R1, R2, R3, R4, R5) da ripartire. L'attuale allocazione e i bisogni massimi sono i seguenti:

Processo	Allocate	Massimo
A	1 0 2 1 1	3 1 2 1 3
B	2 0 1 1 1	3 3 4 2 1
C	1 1 0 1 0	2 1 4 1 0
D	1 1 1 1 0	1 1 3 2 1

Considerando il vettore delle risorse disponibili uguale a [0 1 3 1 2], e utilizzando l'Algoritmo del Banchiere, si discuta se il sistema sia in uno stato sicuro.

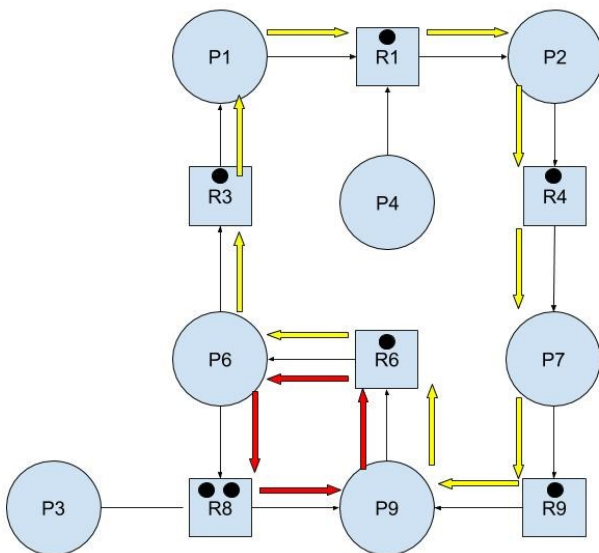
Cognome e nome: _____ Matricola: _____ Posto: _____

Soluzione

Soluzione al Quesito 1

DOMANDA	Vero/Falso
Una system call dà sempre luogo ad un mode switch tra modalità utente e modalità kernel	V
Un process switch tra processi utente avviene sempre contestualmente a 2 mode switch (utente->kernel, kernel->utente)	V
Un interrupt viene gestito in modalità kernel	V
Ogni interrupt è associato ad un processo che ha richiesto una operazione di I/O	F
L'inversione di priorità è una tecnica utilizzata per evitare la <i>starvation</i> dei processi a bassa priorità	F
Un processo per lanciare un nuovo processo deve fare una system call	V

Soluzione al Quesito 2



Sì, c'è uno stallo e il ciclo che lo determina è quello in giallo:

$P1 \Rightarrow R1 \Rightarrow P2 \Rightarrow R4 \Rightarrow P7 \Rightarrow R9 \Rightarrow P9 \Rightarrow R6 \Rightarrow P6 \Rightarrow R3 \Rightarrow P1$

Si noti che $P6 \Rightarrow R8 \Rightarrow P9 \Rightarrow R6 \Rightarrow P6$ (ciclo in rosso) non vale come ciclo per la molteplicità di R8 e la risorsa $R8 \Rightarrow P3$ che potrebbe essere liberata da P3.

Soluzione al Quesito 3

SOLUZIONE [A]: sarà eseguita la print(x) istruzione numero 3 con risultato stampato -1 e poi anche la print(x) istruzione numero 17 con valore stampato -2.

SOLUZIONE [B]: c'è un unico flusso di istruzioni possibili: 8, 9, 10, 11, 12, 1, 2, 3, 4, 5, 13, 14, 15, 16, 17

Può però essere scritto in modi diversi. Es. va bene anche 1, 8, 9, 10, 11, 12, 13, 1, 2, 3, 4, 5, 13, 14, 15, 16, 17 (o simili)

Soluzione al Quesito 4

La matrice delle necessità (massimo numero di risorse richieste dal processo - risorse allocate al processo) è la seguente:

```

2 1 0 0 2
1 3 3 1 0
1 0 4 0 0
0 0 2 1 1
    
```

Il proc. D potrebbe essere eseguito fino alla fine. Quando ha finito, il vettore delle risorse disponibili è [1 2 4 2 2].

Il proc. C potrebbe dunque essere eseguito e al suo completamento, il vettore delle risorse disponibili diverrebbe [2 3 4 3 2].

Questo permetterebbe di eseguire e terminare il processo A ottenendo [3 3 6 4 3] come vettore delle risorse disponibili.

Questo permetterebbe di eseguire anche il processo B.

Il sistema è quindi in uno stato sicuro.