

	Politecnico di Milano Facoltà di Ingegneria dell'Automazione INFORMATICA INDUSTRIALE Appello 13 luglio 2006	COGNOME E NOME	
	RIGA	COLONNA	MATRICOLA

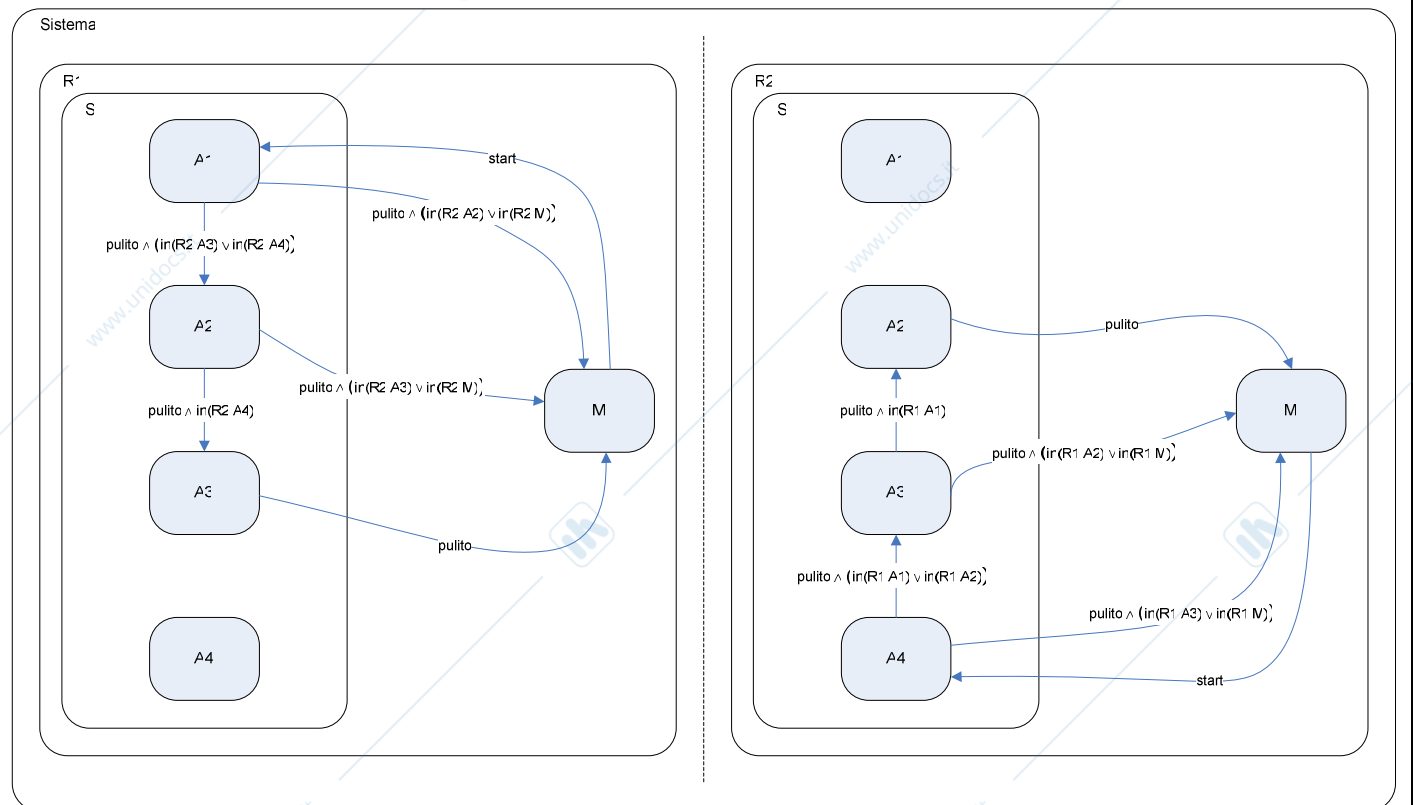
- Il presente plico pinzato, composto di quattro fogli (fronte/retro), deve essere debitamente compilato con cognome, nome, numero di matricola, posizione durante lo scritto, e deve essere firmato.
- I compiti non compilati, non firmati o con fogli mancanti non saranno considerati validi e quindi non saranno corretti.
- Sarà valutato solo quanto scritto su questi fogli.
- Non è consentito consultare testi né appunti.
- Sul tavolo non devono essere presenti telefoni cellulari, né astucci, né custodie di altro tipo.
- Le risposte devono essere scritte negli appositi riquadri, qualsiasi testo esterno a tali riquadri non verrà preso in considerazione.
- Se lo spazio per la soluzione degli esercizi non fosse sufficiente, si può scrivere sull'ultimo foglio.

FIRMA

Esercizio 1 (9 punti). Specificare mediante il formalismo degli *statecharts* senza utilizzare variabili il problema di coordinare due robot pulitori (R1 e R2) per lavare il pavimento di una stanza (S). Viene fornito il funzionamento di ogni robot quando questo lava la stanza da solo, e si richiede di modificare tale funzionamento al fine di permettere ai due robot di lavare il pavimento in modo coordinato. Il funzionamento di ogni robot, quando lava da solo la stanza, è il seguente: si attiva dal magazzino a fronte dell'evento *start*, entra all'interno della stanza, lava la stanza e quando la stanza è completamente pulita torna nel magazzino. Il criterio con cui il robot lava la stanza è il seguente: la stanza (S) viene divisa in quattro aree (A1-A2-A3-A4) contigue e le aree vengono lavate in successione una dopo l'altra (le stanze vengono lavate o nell'ordine A1-A2-A3-A4 o nell'ordine A4-A3-A2-A1). Il passaggio del robot da un'area alla successiva è basato sull'evento *pulito*: questo viene generato ogni qual volta l'area che il robot sta lavando diventi pulita. In sintesi, il robot pulisce un'area finché non si è presentato l'evento *pulito*, e, appena si presenta tale evento, il robot si ferma e si sposta nell'area da pulire successiva (qualora si trovi nell'ultima area, il robot si sposta in magazzino). Le aree possono essere sporche in vario modo richiedendo così tempi diversi per poter essere pulite. Si specifichi il funzionamento coordinato dei due robot in modo tale che:

- R1 e R2 agiscano contemporaneamente,
- R1 inizi a lavare la stanza S partendo dall'area A1 e si muova verso l'area A4,
- R2 inizi a lavare la stanza S partendo dall'area A4 e si muova verso l'area A1,
- tutte le aree vengano lavate una sola volta,
- non siano mai presenti due robot nella stessa area contemporaneamente,
- i tempi di passaggio dei robot tra due aree contigue e tra una generica area e il magazzino siano trascurabili ai fini della specifica.

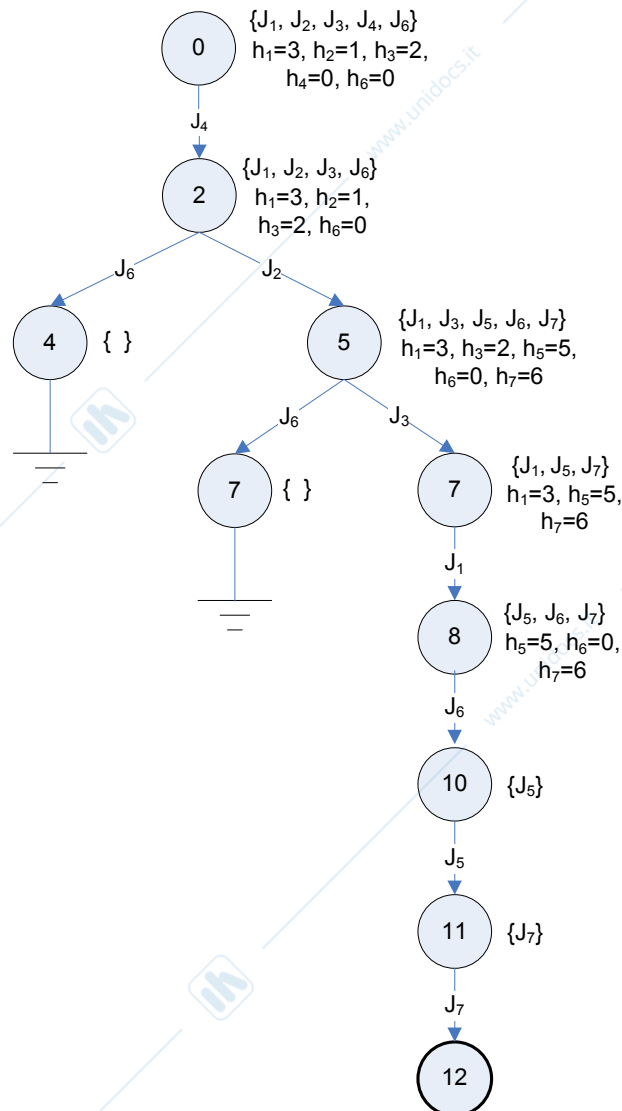
(Suggerimento: si specifichi prima il funzionamento dei robot quando lavano la stanza singolarmente, e poi quando lavano la stanza in modo coordinato.)



Esercizio 2 (7 punti). Si considerino i processi aperiodici riportati nella seguente tabella:

	J_1	J_2	J_3	J_4	J_5	J_6	J_7
a_i	3	1	2	0	5	0	6
C_i	1	3	2	2	1	2	1
d_i	9	5	8	7	11	10	12

Schedularli mediante l'algoritmo di Spring con un solo arco di backtracking, forward checking ed euristica $h_i = a_i$. A parità di altro si scelgano i processi in ordine crescente di pedice. Riportare l'albero di ricerca e riportare in ogni nodo l'assegnamento fatto, i domini generati dal forward checking e le euristiche. Dire se un solo arco di backtracking è sufficiente al fine di trovare una soluzione ammissibile, e in caso affermativo riportare la schedulazione.



L'algoritmo di Spring da luogo a una schedulazione ammissibile, quindi un arco di backtracking è sufficiente.

La schedulazione è $J_4, J_2, J_3, J_1, J_6, J_5, J_7$ e parte all'istante 0 e non prevede interruzioni.

Esercizio 3 (8 punti). Siano dati i processi periodici τ_1 e τ_2 e uno sporadic server SS; i parametri sono riportati nella seguente tabella:

	τ_1	τ_2	SS
ϕ_i	0	2	0
C_i	2	1	2
T_i	8	5	6

Si studi la schedulabilità del problema e successivamente si riporti la schedulazione a fronte dei seguenti processi aperiodici soft real-time gestiti dal SS:

	J_1	J_2	J_3
a_i	1	5	11
C_i	2	1	3

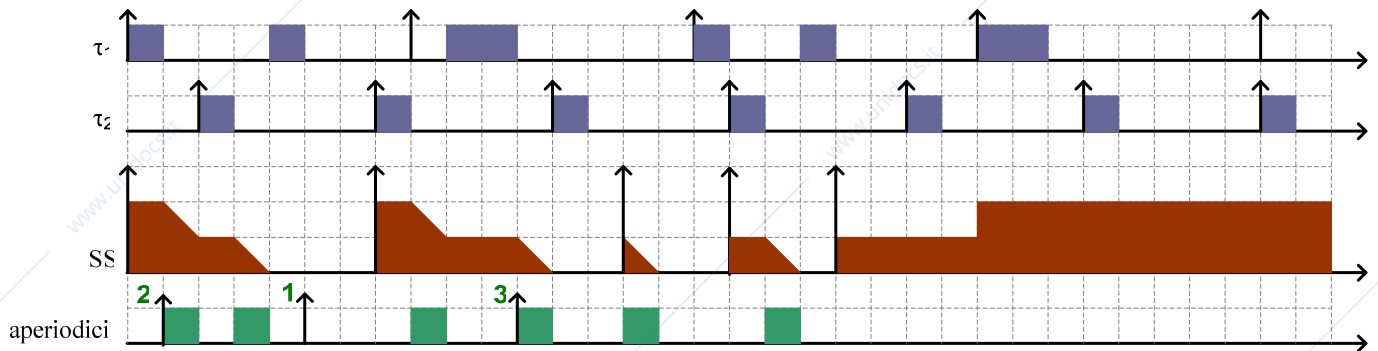
Ai fini dello studio della schedulabilità dei processi, uno sporadic server si comporta esattamente come un processo periodico. Posso quindi studiare la schedulabilità del problema considerato come se si trattassero di tre processi periodici.

Utilizzando Lyu&Layland trovo che U_{lub} è 0.779, mentre il fattore di utilizzazione U è 0.783. Il criterio considerato non mi permette di dire se il problema è schedulabile.

Utilizzando Bini&Buttazzo posso assicurare la schedulabilità del problema, infatti:

$$\left(1 + \frac{1}{4}\right) \cdot \left(1 + \frac{1}{5}\right) \cdot \left(1 + \frac{1}{3}\right) = \frac{5}{4} \cdot \frac{6}{5} \cdot \frac{4}{3} \leq 2$$

Riporto la schedulazione:



Esercizio 4 (7 punti). Si risponda in modo sintetico ed esauriente alle seguenti domande.

1. Spiegare la differenza tra sincronizzazione di primo livello e sincronizzazione di secondo livello.
2. Descrivere la sincronizzazione attraverso "interrupt".

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari

www.unidocs.it - Appunti e dispense per superare i tuoi esami universitari