

PROVA SCRITTA DI SISTEMI IN TEMPO REALE (16/2/2016)

PROBLEMA n. 1

L'esecuzione del seguente insieme di processi periodici, tra loro indipendenti, è affidata ad un sistema di elaborazione monoproiettore:

	T_i [t.u.]	C_i [t.u.]
P_1	4	1
P_2	6	1
P_3	8	2
P_4	12	2

1. Si verifichi se l'insieme di processi è schedabile con l'algoritmo RMPO, ricorrendo all'applicazione, nell'ordine, di uno o più dei seguenti metodi fino ad identificare una risposta conclusiva al riguardo:

- il test di Kuo-Mok

S'	T'_i [t.u.]	C'_i [t.u.]	U'_i
$P'_1 \equiv \{P_1, P_3\}$	4	2	0.500
$P'_2 \equiv \{P_2, P_4\}$	6	2	0.333

$$U = U'_1 + U'_2 = 0.833 > U_{RMPO} (N=2) = 0.828$$

esito positivo	risposta conclusiva								
<table border="1" style="border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">sì</td> <td style="padding: 2px 10px;">no</td> </tr> <tr> <td style="padding: 2px 10px;"></td> <td style="padding: 2px 10px; text-align: center;">×</td> </tr> </table>	sì	no		×	<table border="1" style="border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">sì</td> <td style="padding: 2px 10px;">no</td> </tr> <tr> <td style="padding: 2px 10px;"></td> <td style="padding: 2px 10px; text-align: center;">×</td> </tr> </table>	sì	no		×
sì	no								
	×								
sì	no								
	×								

- il test di Burchard

$$X_1 = 0, X_2 = 0.585, X_3 = 0, X_4 = 0.585, \zeta = 0.585 \times 1 - 1/4 = 0.75$$

$$U = 0.833 > U_{RMPO} (N=4, \zeta=0.585) = 0.767$$

esito positivo	risposta conclusiva								
<table border="1" style="border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">sì</td> <td style="padding: 2px 10px;">no</td> </tr> <tr> <td style="padding: 2px 10px;"></td> <td style="padding: 2px 10px; text-align: center;">×</td> </tr> </table>	sì	no		×	<table border="1" style="border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">sì</td> <td style="padding: 2px 10px;">no</td> </tr> <tr> <td style="padding: 2px 10px;"></td> <td style="padding: 2px 10px; text-align: center;">×</td> </tr> </table>	sì	no		×
sì	no								
	×								
sì	no								
	×								

- il test di Han

S'	T _i ' [t.u.]	C _i ' [t.u.]	U _i '
P ₁ ' ≡ P ₁	4	1	1/4
P ₂ '	4	1	1/4
P ₃ ' ≡ P ₃	8	2	1/4
P ₄ '	8	2	1/4

$$U = U_1' + U_2' + U_3' + U_4' = 1 \leq 1$$

esito positivo		risposta conclusiva	
sì	no	sì	no
×		×	

- l'algoritmo di Audsley

	P ₁	P ₂	P ₃	P ₄
R _i ⁰ = C _i				
R _i ¹ = C _i + I _i (R _i ⁰)				
R _i ² = C _i + I _i (R _i ¹)				
R _i ³ = C _i + I _i (R _i ²)				
R _i ⁴ = C _i + I _i (R _i ³)				

esito positivo		risposta conclusiva	
sì	no	sì	no

- Nell'ipotesi che P₂, P₃ e P₄ abbiano rispettivamente deadline ([t.u.]) D₂ = 2, D₃ = 7 e D₄ = 6, si verifichi se l'insieme di processi è schedabile con l'algoritmo DMPO, ricorrendo all'applicazione, nell'ordine, di uno o più dei seguenti metodi fino ad identificare una risposta conclusiva al riguardo:

- il test basato sul fattore di utilizzazione efficace del processore

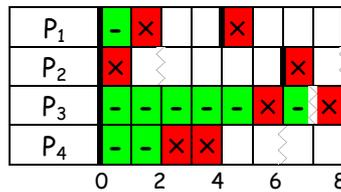
	H _n	H ₁	f	U(n, δ)	f ≤ U(n, δ)
P ₂	{ }	{ }	1/6=0.167	U(1,0.334)=0.334	sì
P ₁	{ }	{P ₂ }	(1+1)/4=0.5	U(1,1)=1	sì
P ₄	{P ₁ }	{P ₂ }	1/4+(2+1)/12=0.5	U(2,0.5)=0.5	no
P ₃	{P ₂ , P ₁ }	{P ₄ }	1/6+1/4+(2+2)/8=0.917	U(3,0.875)=0.74	no

esito positivo		risposta conclusiva	
sì	no	sì	no
	×		×

- il diagramma temporale che evidenzia con riferimento alla seguente notazione

	P _i
	Idle
-	Ready
×	Running

l'evoluzione dei processi dal punto di vista dell'esecuzione nelle condizioni più sfavorevoli



esito positivo

sì	no
	x

risposta conclusiva

sì	no
x	

3. Nelle stesse ipotesi di cui al punto precedente, si verifichi se l'insieme di processi è schedulabile con l'algoritmo EDF ricorrendo all'applicazione, nell'ordine, di uno o più dei seguenti metodi fino ad identificare una risposta conclusiva al riguardo:

- il test basato sulla densità di utilizzazione del processore

	T _i [t.u.]	C _i [t.u.]	D _i [t.u.]	C _i /D _i
P ₁	4	1	4	0.250
P ₂	6	1	2	0.500
P ₃	8	2	7	0.286
P ₄	12	2	6	0.333

$$\Delta = C_1/D_1 + C_2/D_2 + C_3/D_3 + C_4/D_4 = 1.369 > 1$$

esito positivo

sì	no
	x

risposta conclusiva

sì	no
x	

- l'approccio "processor demand"

$$t^* = ((1-2/6)*1+(1-7/8)*2+(1-6/12)*2)/(1-0.833) = 11.5$$

n	BI ⁿ
0	1 + 1 + 2 + 2 = 6
1	2 + 1 + 2 + 2 = 7
2	2 + 2 + 2 + 2 = 8
3	2 + 2 + 2 + 2 = 8

$$BI = 8$$

$$\min \{t^*, BI\} = 8$$

$$\mathcal{D}^* \cap \mathcal{D} = \{2, 4, 6, 7\}$$

t	$C_1(0,t)$	$C_2(0,t)$	$C_3(0,t)$	$C_4(0,t)$	$C_p(0,t)$	$\leq t$
2	0	1	0	0	1	sì
4	1	1	0	0	2	sì
6	1	1	0	2	4	sì
7	1	1	2	2	6	sì

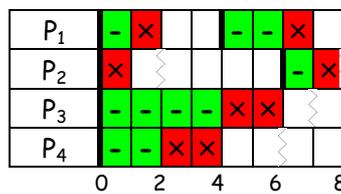
esito positivo

sì	no
×	

risposta conclusiva

sì	no
×	

4. Nelle stesse ipotesi di cui al punto precedente, si verifichi tramite la costruzione di un diagramma temporale se l'insieme di processi è schedabile con l'algoritmo non strict LST:



slack	t=0	t=4	t=6
P ₁	3	3	1
P ₂	1	-	1
P ₃	5	1	-
P ₄	4	-	-

esito positivo

sì	no
×	

risposta conclusiva

sì	no
×	

PROBLEMA n. 2

Un sistema di elaborazione monoprocesso ha il compito di eseguire quattro processi periodici P_1, P_2, P_3, P_4 tra loro indipendenti, gestendo nel contempo quattro richieste aperiodiche $R_{a1}, R_{a2}, R_{a3}, R_{a4}$:

	T_i [t.u.]	C_i [t.u.]
P_1	8	2
P_2	10	1
P_3	12	2
P_4	15	2

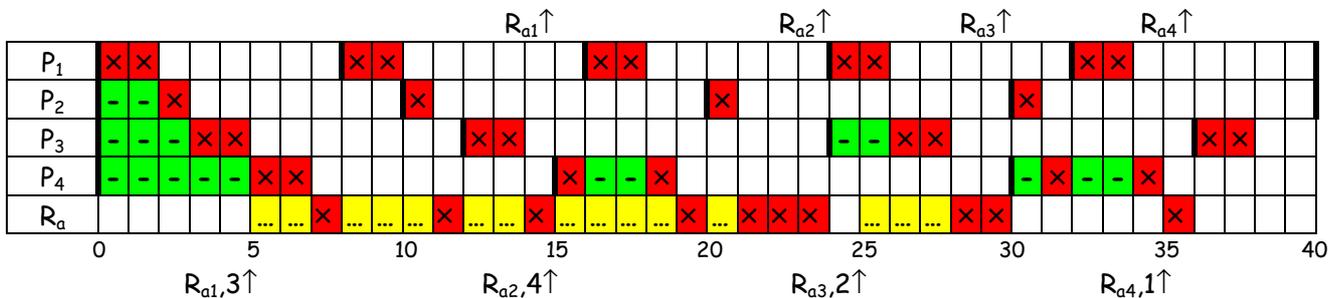
	a_i [t.u.]	C_i [t.u.]
R_{a1}	5	3
R_{a2}	15	4
R_{a3}	25	2
R_{a4}	35	1

A) Con l'ausilio di diagrammi temporali e facendo riferimento alla seguente notazione

	P_i	R_a	S
	Idle	None	Idle
-	Ready	-	Ready
...	-	Pending	Waiting
x	Running	Being Served	Running

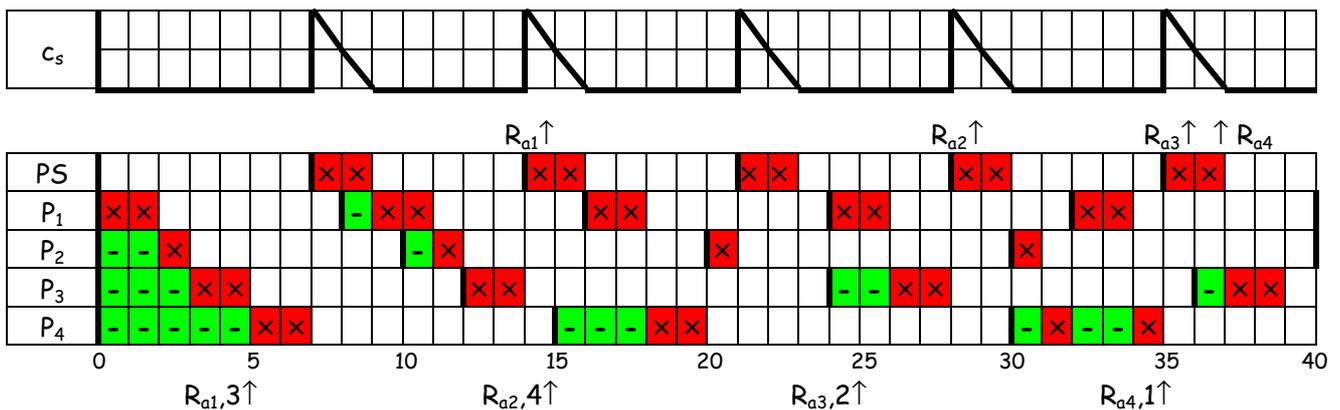
si determini (riportandone i valori nella sottostante tabella) il tempo di completamento della esecuzione f_i [t.u.] ed il ritardo sul tempo di risposta $f_i - a_i - C_i$ [t.u.] delle richieste aperiodiche derivanti dalla loro gestione tramite

1. Servizio in Background (strategia di schedulazione dei processi: RMPO)

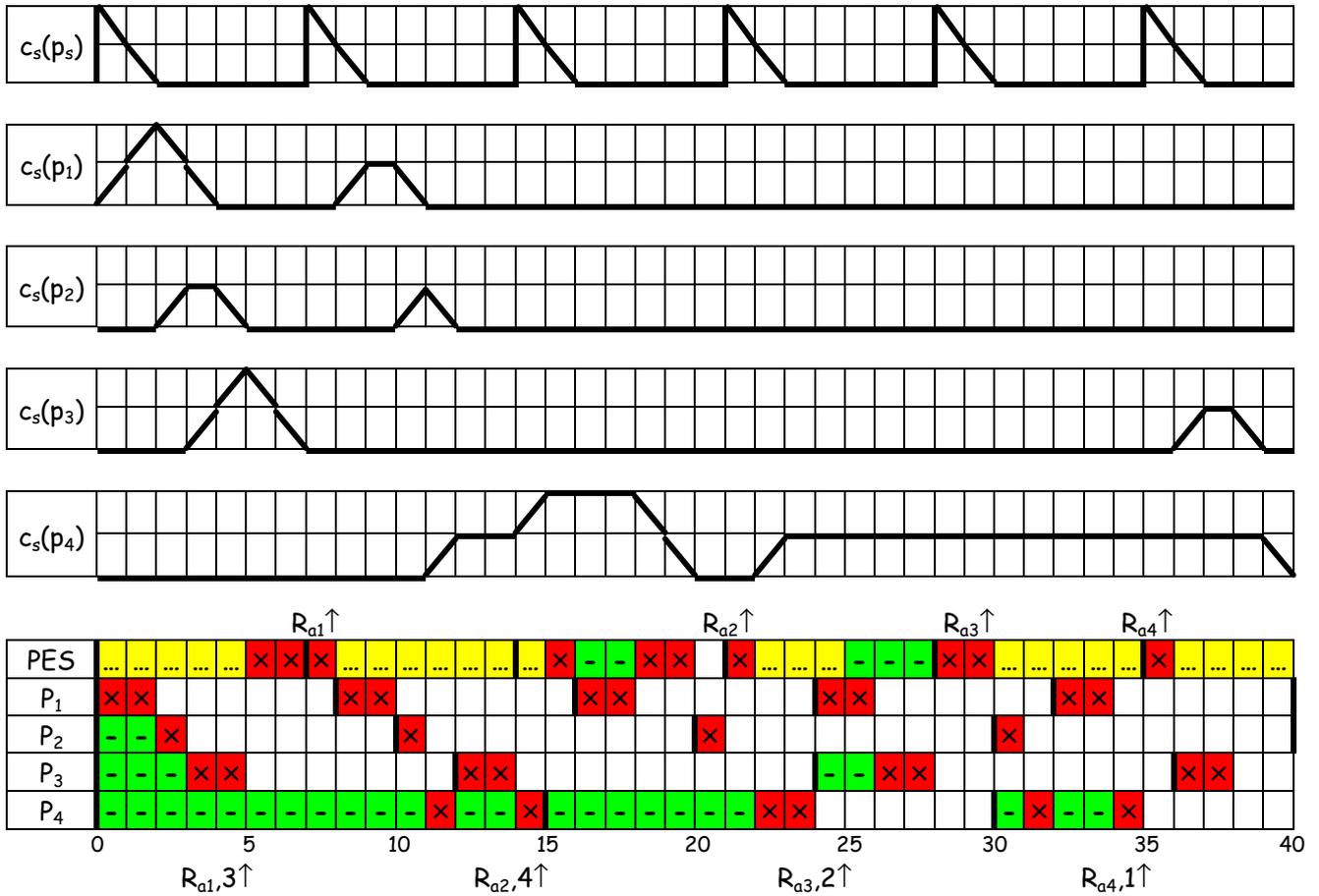


2. Server a priorità statica di periodo $T_s = 7$ t.u. e capacità $C_s = 2$ t.u. (strategia di schedulazione dei processi e del Server: RMPO)

• Polling Server



- Priority Exchange Server

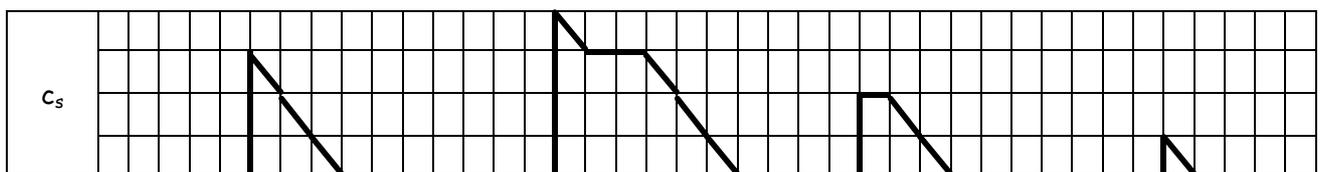


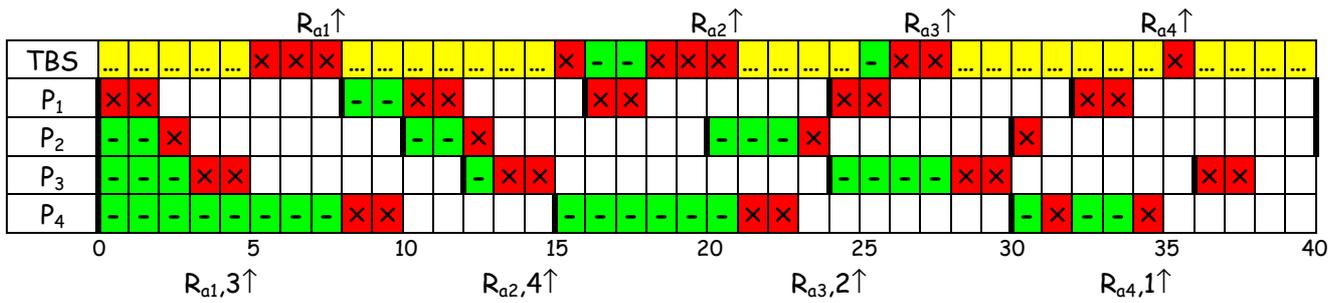
3. Server a priorità dinamica con fattore di utilizzazione del processore U_s più elevato possibile compatibilmente con la schedulabilità dei processi periodici (strategia di schedulazione dei processi e del Server: EDF)

$$U_s = 1 - U_p = 1 - 0.65 = 0.35$$

- Total Bandwidth Server

	a_i [t.u.]	C_i [t.u.]	d_{si} [t.u.]
R_{a1}	5	3	$5 + 3 / 0.35 = 13.6$
R_{a2}	15	4	$15 + 4 / 0.35 = 26.4$
R_{a3}	25	2	$26.4 + 2 / 0.35 = 32.1$
R_{a4}	35	1	$35 + 1 / 0.35 = 37.9$





• Constant Utilization Server

Si verifichi se CUS è in grado di fornire le stesse prestazioni di TBS, motivandone le ragioni ed evidenziando in caso contrario nella tabella sottostante i dati relativi alle richieste aperiodiche il cui tempo di completamento del servizio risulta superiore

$$\begin{aligned}
 R_{a1}: & & f_1(CUS) &= f_1(TBS) \\
 R_{a2}: a_2 = 15 > d_{s1} = 13.6 & \rightarrow & f_2(CUS) &= f_2(TBS) \\
 R_{a3}: a_3 = 25 < d_{s2} = 26.4 & \rightarrow & f_3(CUS) &> f_3(TBS) \\
 R_{a4}: a_4 = 35 > d_{s3} = 32.1 & \rightarrow & f_4(CUS) &= f_4(TBS)
 \end{aligned}$$

	a_i	C_i	f_i	$f_i - a_i - C_i$								
R_{a1}	5	3	15	7	15	7	8	0	8	0		
R_{a2}	15	4	24	5	29	10	22	3	21	2		
R_{a3}	25	2	30	3	36	9	30	3	28	1	28.4	1.4
R_{a4}	35	1	36	0	37	1	36	0	36	0		
			BKG		PS		PES		TBS		CUS	

B) Si verifichi se, indipendentemente dalla effettiva distribuzione temporale e dal tempo di servizio delle richieste aperiodiche, la loro gestione tramite PS e PES compromette la schedulabilità

(a) dei processi P_1, P_2, P_3, P_4 ,

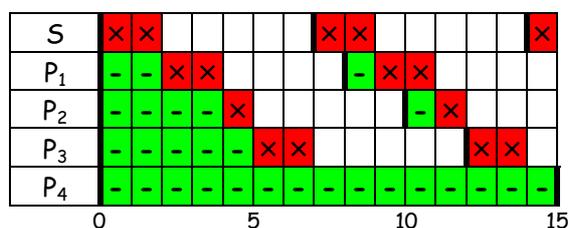
(b) di qualunque altro insieme di 4 processi periodici P'_1, P'_2, P'_3, P'_4 aventi complessivamente lo stesso fattore di utilizzazione del processore ($U'_1 + U'_2 + U'_3 + U'_4 = U_1 + U_2 + U_3 + U_4$) e periodi $T'_1, T'_2, T'_3, T'_4 > T_s$,

identificando in entrambi i casi il valore massimo che C_s può assumere, a parità di T_s , onde prevenire tale inaccettabile circostanza.

• (a)

sì	no
×	

infatti:



$$C_s \leq 1 \text{ t.u.}$$

- (b)

sì	no
×	

infatti:

$$U_s = 0.286 > 2 / ((1 + 0.65 / 4)^4) - 1 = 0.095$$

$$C_s \leq 7 * 0.095 = 0.66 \text{ t.u.}$$

PROBLEMA n. 3

L'esecuzione concorrente del seguente insieme di processi

	T_i [t.u.]	D_i [t.u.]	C_i [t.u.]
P_1	20	20	5
P_2	25	25	6
P_3	35	30	5
P_4	45	45	4
P_5	50	50	6

attivati all'istante ϕ_i ([t.u.]) = 8, 6, 4, 2, 0, rispettivamente, è affidata ad un sistema di elaborazione monoproiettore.

I processi P_1, P_2, P_3 e P_5 condividono 4 risorse R_1, R_2, R_3 e R_4 , con accesso mutuamente esclusivo ($u_1 = u_2 = u_3 = u_4 = 1$) regolamentato dai semafori S_1, S_2, S_3 e S_4 . Più precisamente, l'esecuzione di ciascun job di ogni processo comporta le seguenti elaborazioni, contraddistinte dal tempo indicato tra parentesi:

- P_1 : esecuzione normale (1 t.u.), accesso a R_3 (1 t.u.), accesso a R_2 (1 t.u.), accesso a R_1 (1 t.u.), esecuzione normale (1 t.u.);
- P_2 : esecuzione normale (1 t.u.), accesso a R_1 (4 t.u. complessivamente) con accesso annidato dopo 2 t.u. a R_4 (1 t.u.), esecuzione normale (1 t.u.);
- P_3 : esecuzione normale (1 t.u.), accesso a R_4 (4 t.u. complessivamente) con accesso annidato dopo 2 t.u. a R_2 (1 t.u.);
- P_4 : esecuzione normale (4 t.u.);
- P_5 : esecuzione normale (1 t.u.), accesso a R_2 (4 t.u. complessivamente) con accesso annidato dopo 2 t.u. a R_3 (1 t.u.), esecuzione normale (1 t.u.).

	C_{type}				
P_1		R_3	R_2	R_1	
P_2		R_1	R_1	R_4	R_1
P_3		R_4	R_4	R_2	R_4
P_4					
P_5		R_2	R_2	R_3	R_2

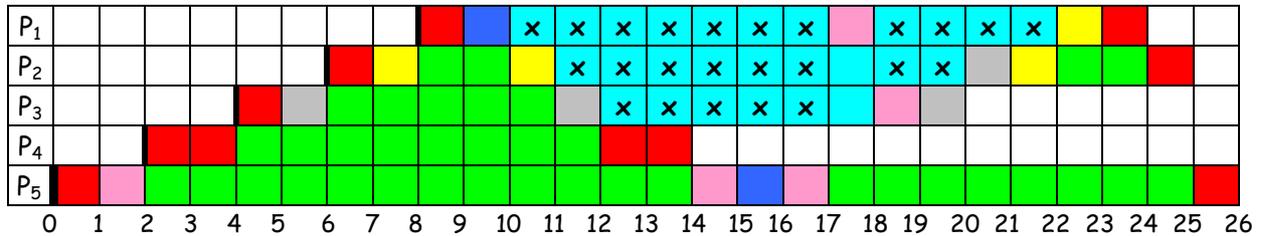
1. Nell'ipotesi che i processi siano schedulati in accordo alla strategia DMPO,

➤ si identifichi con l'ausilio di diagrammi temporali, facendo riferimento alla seguente notazione

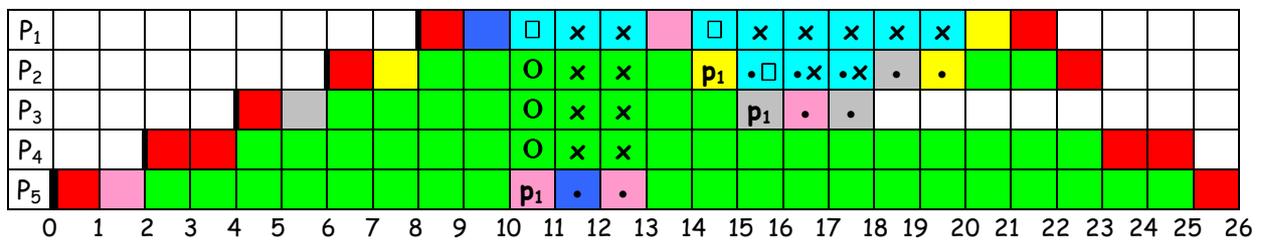
	P_i
	Idle
	ready ($\times [O, \nabla]$ when blocked)
	suspended ($\times [\square, \nabla]$ when blocked)
	running without locking any semaphore
	executing critical section protected by semaphore S_1
	executing critical section protected by semaphore S_2
	executing critical section protected by semaphore S_3
	executing critical section protected by semaphore S_4

ed evidenziando esplicitamente la priorità corrente (π_i) di un processo soltanto se essa differisce dalla corrispondente priorità nominale (p_i), il tempo di completamento dell'esecuzione (f_i [t.u.]) ed il ritardo sul tempo di risposta ($f_i - \phi_i - C_i$ [t.u.]) del primo job di ciascun processo (riportandone i valori nella sottostante tabella) nel caso di:

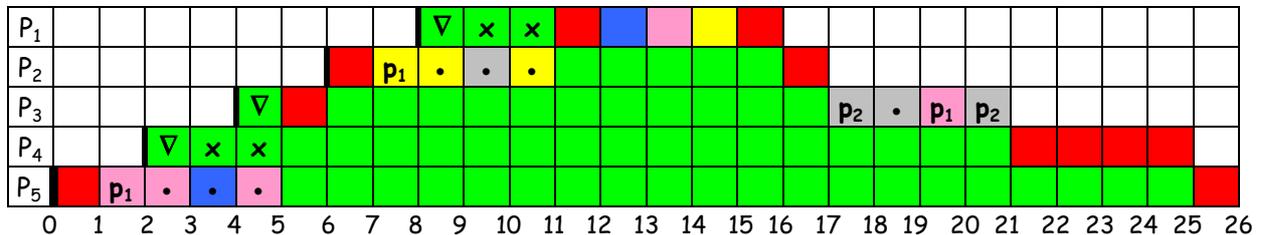
- esecuzione in base alle relative priorità statiche (NOP)



- applicazione del protocollo priority inheritance (PIP)



- applicazione del protocollo immediate priority ceiling (IPCP)



	ϕ_i	C_i	f_i	$f_i - \phi_i - C_i$	f_i	$f_i - \phi_i - C_i$	f_i	$f_i - \phi_i - C_i$
P_1	8	5	24	11	22	9	16	3
P_2	6	6	25	13	23	11	17	5
P_3	4	5	20	11	18	9	21	12
P_4	2	4	14	8	25	19	25	19
P_5	0	6	26	20	26	20	26	20
	NOP				PIP		IPCP	

➤ Si determini il massimo tempo di blocco di ogni processo derivante dall'applicazione di PIP, IPCP

	R_1	R_2	R_3	R_4	B_i	
P_1	1	1	1		12	4
P_2	4			1	8	4
P_3		1		4	4	4
P_4					4	4
P_5		4	1		0	0
					PIP	IPCP

e si verifichi, applicando l'algoritmo di Audsley, se l'insieme dei processi è schedulabile qualunque sia il protocollo usato.

	P ₁	P ₂	P ₃	P ₄	P ₅
R _i ⁰	5 + 12 = 17	6 + 8 = 14	5 + 4 = 9	4 + 4 = 8	6 + 0 = 6
R _i ¹	17 + 0 = 17	14 + 5 = 19	9 + 5 + 6 = 20	8 + 5 + 6 + 5 = 24	6 + 5 + 6 + 5 + 4 = 26
R _i ²		14 + 5 = 19	9 + 5 + 6 = 20	8 + 10 + 6 + 5 = 29	6 + 10 + 12 + 5 + 4 = 37
R _i ³				8 + 10 + 12 + 5 = 35	6 + 10 + 12 + 10 + 4 = 42
R _i ⁴				8 + 10 + 12 + 5 = 35	6 + 15 + 12 + 10 + 4 = 47
R _i ⁵					6 + 15 + 12 + 10 + 8 = 51
R _i ⁶					6 + 15 + 18 + 10 + 8 = 57
R _i ⁷					6 + 15 + 18 + 10 + 8 = 57

	R _i [t.u.]	D _i [t.u.]	R _i ≤ D _i
P ₁	17	20	sì
P ₂	19	25	sì
P ₃	20	30	sì
P ₄	35	45	sì
P ₅	57	50	no

esito positivo

sì	no
	×

risposta conclusiva

sì	no
×	

2. Nell'ipotesi che (a) i processi siano schedulati in accordo alla strategia EDF, (b) gli accessi alle risorse condivise siano gestiti secondo il protocollo stack resource policy (SRP), (c) il numero di unità disponibili per ogni tipologia di risorsa sia rispettivamente $u_1 = 3$, $u_2 = 3$, $u_3 = 1$, $u_4 = 1$, (d) il numero di unità richiesto da ciascun processo per ogni tipologia di risorsa sia

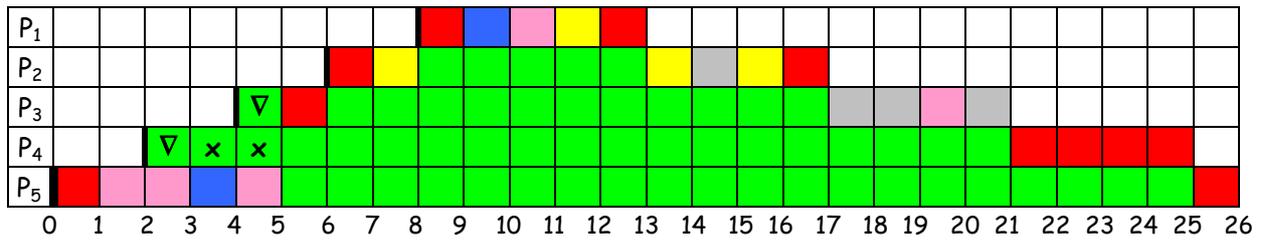
	R ₁	R ₂	R ₃	R ₄
P ₁	2	1	1	
P ₂	1			1
P ₃		2		1
P ₄				
P ₅		3	1	

- si individui il livello di preemption dei processi ed il tetto di preemption che contraddistingue ciascuna risorsa al variare del numero di unità contestualmente libere

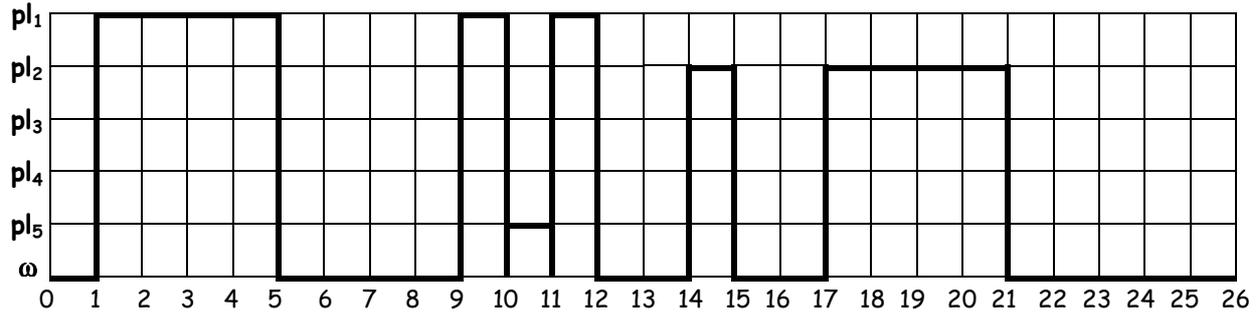
P ₁	pl ₁ (max)
P ₂	pl ₂
P ₃	pl ₃
P ₄	pl ₄
P ₅	pl ₅ (min)

	v _k =0	v _k =1	v _k =2	v _k =3
R ₁	pl ₁	pl ₁	∞	∞
R ₂	pl ₁	pl ₃	pl ₅	∞
R ₃	pl ₁	∞	-	-
R ₄	pl ₂	∞	-	-

- si identifichi con l'ausilio di un diagramma temporale il tempo di completamento dell'esecuzione del primo job di ciascun processo



$\Pi\Lambda_s$



	ϕ_i	C_i	f_i	$f_i - \phi_i - C_i$
P ₁	8	5	13	0
P ₂	6	6	17	5
P ₃	4	5	21	12
P ₄	2	4	25	19
P ₄	0	6	26	20
SRP				

- si determini il massimo tempo di blocco di ogni processo

	R ₁	R ₂	R ₃	R ₄	B _i
P ₁	1	1	1		4
P ₂	4			1	4
P ₃		1		4	4
P ₄					4
P ₅		4	1		0

- si verifichi se l'insieme dei processi è schedulabile

$P_1 : C_1 / D_1 + B_1 / D_1 = 0.45 \leq 1$ OK
 $P_2 : C_1 / D_1 + C_2 / D_2 + B_2 / D_2 = 0.65 \leq 1$ OK
 $P_3 : C_1 / D_1 + C_2 / D_2 + C_3 / D_3 + B_3 / D_3 = 0.79 \leq 1$ OK
 $P_4 : C_1 / D_1 + C_2 / D_2 + C_3 / D_3 + C_4 / D_4 + B_4 / D_4 = 0.83 \leq 1$ OK
 $P_5 : C_1 / D_1 + C_2 / D_2 + C_3 / D_3 + C_4 / D_4 + C_5 / D_5 = 0.86 \leq 1$ OK

esito positivo	
sì	no
×	

risposta conclusiva	
sì	no
×	