

### Soluzione esercizio 1

Si usa algoritmo del banchiere perché ci sono più istanze di ogni tipo di risorsa, la situazione iniziale è

	Alloc				Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	0	0	0	0	1	1	1	1	2	2	2	2
P2	0	0	0	0	1	1	1	1				
P3	0	0	0	0	1	1	1	1				
P4	0	0	0	0	1	1	1	1				

A T1 P1 richiede A e B che sono disponibili e portano in stato:

	Alloc				Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	1	1	0	0	0	0	1	1	1	1	2	2
P2	0	0	0	0	1	1	1	1				
P3	0	0	0	0	1	1	1	1				
P4	0	0	0	0	1	1	1	1				

Che è uno stato sicuro (seq. sicura P1, P2, P3, P4)

A T2 P2 richiede B, C e D che sono disponibili e portano nello stato:

	Alloc				Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	1	1	0	0	0	0	1	1	1	0	1	1
P2	0	1	1	1	1	0	0	0				
P3	0	0	0	0	1	1	1	1				
P4	0	0	0	0	1	1	1	1				

Che è uno stato sicuro (seq. sicura P1, P2, P3, P4)

A T3 P3 richiede A, Nb B, Nc C e Nd D. Nb necessariamente è 0 perché non sono disponibili risorse B quindi le possibili richieste sono (1, 0, 0, 0), (1, 0, 0, 1), (1, 0, 1, 0) e (1, 0, 1, 1)

Nel primo caso si ha:

	Alloc				Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	1	1	0	0	0	0	1	1	0	0	1	1
P2	0	1	1	1	1	0	0	0				
P3	1	0	0	0	0	1	1	1				
P4	0	0	0	0	1	1	1	1				

Che è uno stato sicuro (seq. sicura P1, P2, P3, P4)

Nel secondo caso si ha:

	Alloc				Need				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	1	1	0	0	0	0	1	1	0	0	1	0
P2	0	1	1	1	1	0	0	0				
P3	1	0	0	1	0	1	1	0				
P4	0	0	0	0	1	1	1	1				

Non ci sono sequenze sicure perché le risorse disponibili non permettono di far terminare nessun processo. Cosa analoga accade anche per gli altri due casi.

Quindi solo se  $N_b=N_c=N_d=0$  si permane in uno stato sicuro.