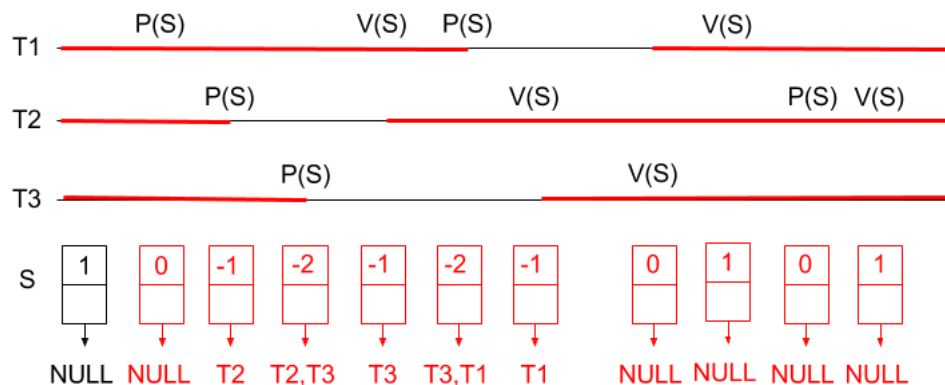


Esercizio 1

Considerare tre thread, in esecuzione parallela su 3 core, che si sincronizzano tramite il semaforo S inizializzato a 1, a protezione di una sezione critica:



Indicare nello schema sopra lo stato del semaforo S (valore e coda) in corrispondenza di ogni P e V effettuata dai thread e evidenziare i thread in esecuzione (tramite una riga più grossa).

Breve spiegazione (indicare se i thread sono bloccati): E' un classico semaforo mutex inizialmente verde: la prima P non è bloccante e rende il semaforo rosso. Le successive due P sono bloccanti e mettono i thread T2 e T3 in coda. La successiva V sblocca il thread T2 che torna in esecuzione. Il thread T1 si blocca eseguendo una P e si mette in coda (dopo T3). La successiva V eseguita da T2 sblocca T3 il quale sblocca T1. La V eseguita da T1 rende nuovamente il semaforo verde (nessuno è in una sezione critica). Le successive P e V eseguite da T2 rendono il semaforo rosso e poi nuovamente verde.

Esercizio 2

Considerare la seguente proposta (non corretta) di soluzione al problema della sezione critica:

```

boolean locked = false;

thread T0 {
    while(locked) {}
    locked = true;
    < sezione critica >
    locked = false;
}

thread T1 {
    while(locked) {}
    locked = true;
    < sezione critica >
    locked = false;
}
    
```

- Mostrare un esempio di esecuzione in cui non c'è mutua esclusione (T0 e T1 sono in sezione critica):
 Supponiamo che T0 stia entrando in sezione critica e abbia superato il while in quanto locked è false. Se anche T1 cerca di entrare in sezione critica prima che T0 abbia messo locked a true, T1 supererà il ciclo while e entrambi i thread saranno in sezione critica. Questo rompe la mutua esclusione che richiede, appunto, che T0 e T2 non siano in sezione critica.
- Può succedere che T0 e T1 entrino in stallo (attesa circolare)? Motivare la risposta:
 L'unica istruzione "bloccante" è il ciclo while. Supponiamo che T0 sia bloccato nel ciclo while. Questo significa che locked è true. L'unico altro thread che può aver impostato locked a true è T1 ma in tale caso T1 sarà in sezione critica e prima o poi imposterà locked a false liberando T0. Lo stesso vale se supponiamo che T1 sia bloccato e T0 in sezione critica.