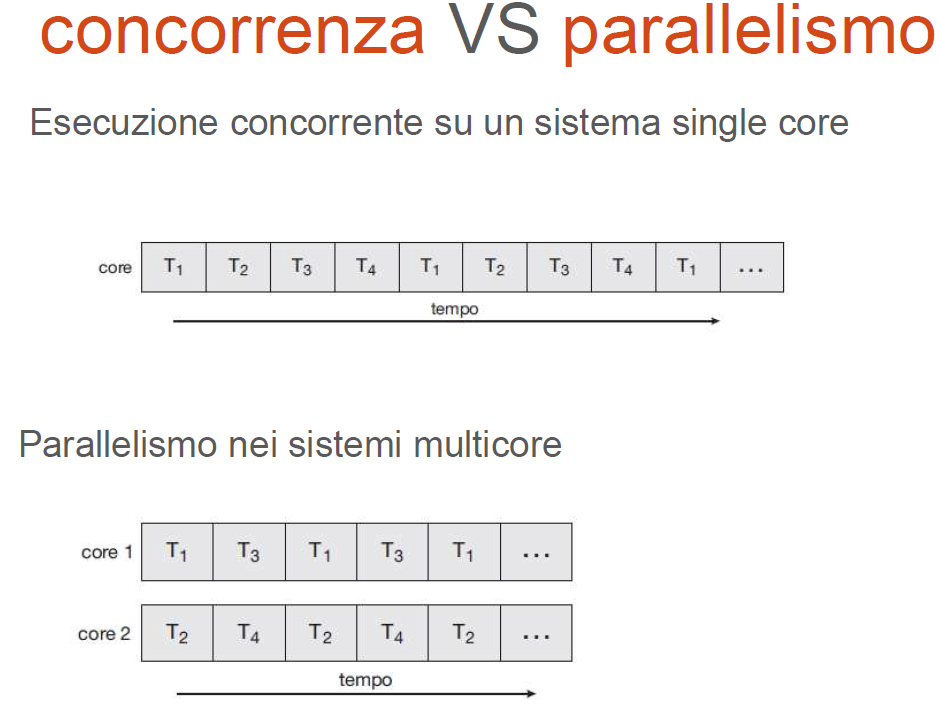
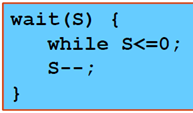
***SINCRONIZZAZIONE DEI PROCESSI***



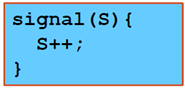
In un *sistema monoprocessore* è sufficiente interdire le interruzioni mentre si modificano le *variabili condivise* (valori che sono modificati da stessi processi).

I ***semafori*** sono strumenti di sincronizzazione.

* Si può accedere al semaforo *S* (dopo l’inizializzazione) solo attraverso due operazioni atomiche predefinite *wait( )* e *signal( );*



Nel ***wait( )*** devono essere effettuate sia la ***verifica del valore*** del semaforo che il suo ***decremento***.



Nel ***signal( )*** viene effettuato solo l’***incremento*** del semaforo.

* Tutte le modifiche al valore del semaforo contenute nelle operazioni di *wait( )* e *signal(* *)*, devono essere eseguite in modo indivisibile;
* Mentre un processo cambia il valore del semaforo, nessun altro processo può contemporaneamente modificare quello stesso valore;
* Nel *wait( )* devono essere effettuate sia la verifica del valore del semaforo che il suo decremento;
* ***Semaforo binario***: assume soltanto i valori 0 e 1, altrimenti è un ***Semaforo contatore***!
* Quando il semaforo vale =*0*, tutte le istanze della risorsa sono allocate e i processi che le richiedono devono *sospendersi* *sul semaforo* fino a che esso ridiventa positivo!