Interazione tra processi

Elaborazione parallela

Interazioni indirette

- I processi concorrenti, anche se operano su variabili separate, e quindi ognuno ignora la presenza dell'altro, possono interagire indirettamente perché richiedono l'uso di una stessa risorsa.
- Il sistema operativo detiene risorse limitate da assegnare ai processi in esecuzione.

La mutua esclusione

- Il problema della mutua esclusione riguarda il caso generale dei conflitti tra i processi che si contendono l'uso di una stessa risorsa.
- Il problema della mutua esclusione può essere descritto nel modo seguente:
- Il sistema operativo possiede una risorsa R e gestisce i processi P_1, P_2, … P_n. Assicurare che in ogni istante vi sia un solo processo autorizzato ad usare la risorsa R e che ogni altro processo P_i che chiederà la risorsa, la possa utilizzare entro un tempo finito.

La mutua esclusione

Una soluzione efficiente, al problema della mutua esclusione, deve possedere i seguenti requisiti:

1. in ogni istante la risorsa è usata da un solo processo,
2. se più processi richiedono simultaneamente l'uso di una risorsa, solo uno di essi sarà autorizzato, entro un tempo finito, a usare la risorsa,
3. un processo che ha ottenuto una risorsa deve usarla e poi rilasciarla entro un tempo finito,
4. un processo non deve consumare tempo di elaborazione quando è in *attesa* di acquisire la risorsa.

Per soddisfare il IV requisito, il sistema operativo sposta il processo dallo stato di esecuzione allo stato di *attesa della risorsa*, dal quale verrà rimosso quando la risorsa si renderà disponibile. A questa operazione provvede il sistema operativo gestendo la lista dei processi in attesa.

Accesso ad una risorsa

Un processo che deve usare una risorsa, non consentendo ad altri processi di interferire con le sue operazioni, si comporta nella seguente maniera:

1. interroga lo stato, libera o assegnata, della risorsa,
2. quando la trova libera se ne riserva l'uso, altrimenti passa in stato di attesa,
3. usa la risorsa,
4. rilascia la risorsa.

Primitive di sincronizzazione

Una risorsa fisica condivisa da più processi si può trovare o nello stato *libera* oppure nello stato *assegnata*, e quindi è rappresentabile con una variabile (booleana o anche intera) accessibile, in mutua esclusione, a più processi. Una tale variabile è detta di tipo **semaforo**.

Il processo P per richiedere una risorsa esegue le due azioni:

1. Legge lo stato di R
2. Se R è libera pone R nello stato “Assegnata”

Potrebbe accadere che, tra la prima e la seconda operazione, scada il quanto di tempo. Il processo che subentra richiede l’uso della stessa risorsa, quindi esegue le due azioni 1: interroga R e 2: riserva R, poiché P non ha ancora modificato lo stato della risorsa, il processo Q se ne riserva l’uso. Quando P tornerà in esecuzione riprenderà dalla seconda operazione, violando la mutua esclusione, si riserva anche’esso l’uso della risorsa.

Il sistema operativo deve trattare le due azioni come una sola istruzione, e tra esse non deve intercalare altre azioni e non deve servire interruzioni.

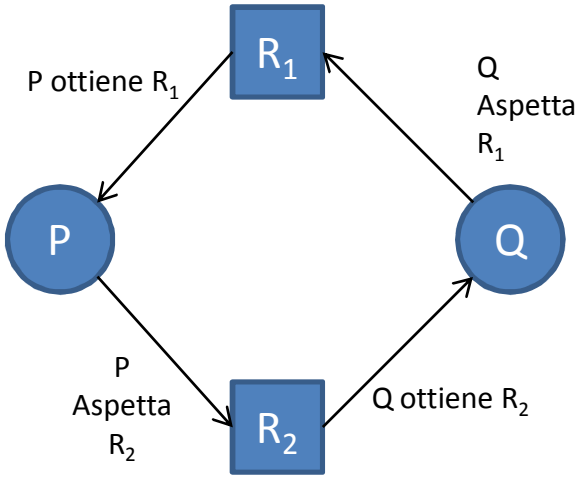
Grafo di Holt

- Le risorse sono rappresentate con quadrati
- I processi sono rappresentati con cerchi
- Un arco parte da un processo e termina su una risorsa se il processo deve aspettare che la risorsa diventi libera.
- Un arco parte da una risorsa e termina su un processo se la risorsa è assegnata al processo.

Stallo di processi

Si considerino due semafori R_1 e R_2, che realizzano la mutua esclusione nell'accesso a due risorse del sistema, e due processi P e Q.

Si supponga che il processo P chieda e ottenga l'uso di R_1, e che il processo Q chieda e ottenga l'uso di R_2, poi, mentre prosegue l'esecuzione parallela dei due processi, prima che rilascino le risorse, si supponga che si verifichino le richieste complementari, cioè che il processo P chieda l'uso di R_2, la quale essendo occupata da Q, provocherà il passaggio del processo P nello stato di attesa dell'evento: "R_2 libera", e che il processo Q chieda l'uso di R_1, la quale essendo assegnata a P, provocherà il passaggio di Q nello stato di attesa dell'evento: "R_1 libera".



Lo stallo si riconosce, sul grafo di Holt, dalla presenza di un ciclo.

Stallo

I due processi non potranno più tornare in esecuzione, si trovano in una situazione di **stallo**, cioè si trovano nello stato di attesa di un evento che non potrà mai accadere.

Non esistono tecniche per uscire da una situazione di stallo, ma è possibile prevenirlo.

L'esempio precedente dei processi P e Q, che restano in attesa reciproca che l'altro rilasci una risorsa, nella sua semplicità, contiene tutti i fattori che sono causa di uno stallo, essi sono:

1. la risorsa è soggetta al meccanismo della mutua esclusione,
2. la risorsa può essere rilasciata solo dal processo che la ha acquisita,
3. a un processo è consentito di acquisire di volta in volta le risorse di cui ha bisogno,
4. si può creare una coda circolare in cui ciascun processo resta in attesa di una risorsa assegnata a un altro processo.

Lo stallo dei processi viene prevenuto assicurando che almeno una delle condizioni appena elencate non si verifichi mai.

Prevenzione dello stallo

- Non rispettare la prima condizione comporta che una risorsa venga usata contemporaneamente da più processi. Ad esempio su una stampante si troverebbero parti stampate da un processo confuse con le parti stampate da un altro processo.
- Se si forzasse un processo a rilasciare temporaneamente la risorsa, prima di avere finito di usarla, gli utenti dovrebbero provvedere a controllare l'inserimento dei propri supporti esterni (dischi, fogli da stampare ecc...). Questa tecnica è usata per quei casi in cui è realizzabile il multiplexing delle risorse, cioè l'uso turnificato di una risorsa da parte di più processi.
- Per evitare la terza condizione, un processo deve riservarsi in anticipo tutte le risorse di cui ha bisogno. Le conseguenze di tale metodo si riflettono sul rendimento globale del sistema, in quanto esiste una più alta probabilità che i processi trovino le risorse occupate, anche se il processo che le ha impegnate le usa per un tempo breve rispetto alla sua durata.

Prevenzione dello stallo

per prevenire lo stallo si deve di stabilire un ordine per soddisfare le richieste, evitando l'attesa circolare di processi che aspettano che un altro rilasci la risorsa.

Processo P_1	Processo P_2	Processo P_3
usa R_1 aspetta R_2	usa R2 aspetta R_3	usa R_3 aspetta R_1

Il processo P_1 ha ottenuto l'uso della risorsa R_1 ma, per continuare ha bisogno della risorsa R_2, trattenuta da P_2, a sua volta P_2 è in attesa di una risorsa trattenuta da P_3, che a sua volta è in attesa che P_1 rilasci la risorsa R_1. Si è formata così una coda di attesa circolare dalla quale nessun processo potrà uscire per passare in esecuzione.

Algoritmo del banchiere

- L’algoritmo da applicare deve esaminare le richieste di assegnazione delle risorse presentate dai processi, e deve soddisfare quelle richieste che consentono al processo di completare le sue operazioni, che quindi potrà rilascerà le risorse usate. Per individuare il processo da favorire, il sistema operativo deve conoscere in anticipo le risorse che i processi richiederanno. Il costo da pagare per prevenire lo stallo, evitando che si formi l’attesa circolare di processi, introduce un lungo tempo di calcolo per determinare l'ordine con cui soddisfare le richieste di uso delle risorse.

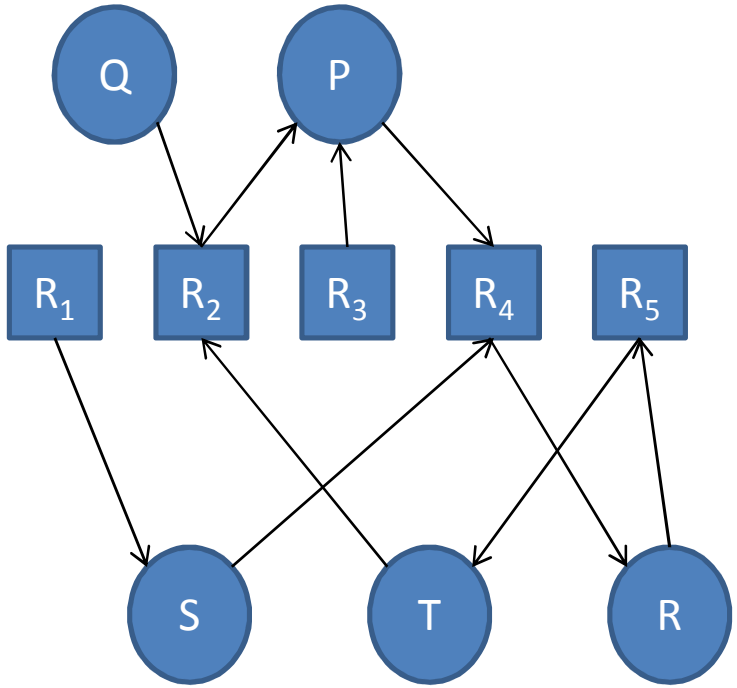
Esempio

Un sistema di calcolo detiene 5 risorse (R_1, R_2, R_3, R_4, R_5).

5 processi (P, Q, R, S, T) in esecuzione si contendono l'uso di queste risorse.

Si consideri la seguente sequenza di richieste:

1. P richiede e ottiene R_2
2. P richiede e ottiene R_3
3. R richiede e ottiene R_4
4. P richiede R_4 (aspetta R_4)
5. Q richiede R_2 (aspetta R_2)
6. T richiede e ottiene R_5
7. S richiede e ottiene R_1
8. R richiede R_5 (aspetta (R_5)
9. T richiede R_2 (aspetta R_2)
10. S richiede R_4 (aspetta R_4)



Dopo l'ultima richiesta il sistema entra in stallo.

soluzione

- Il sistema è in stallo, infatti c'è un ciclo P, R_4, R, R_5, T, R_2, P che corrisponde ad una coda circolare tra i processi P, R, T.
- Inoltre sono in stallo anche S che attende R_4 (assegnata a R) e Q che attende R_2 (assegnata a P)
- Sarebbe sufficiente far terminare uno qualsiasi tra P, R o T.
- Terminare S o Q non elimina lo stallo.