

Contents

1 Deadlocks

Facciamo un esempio stupido Devo prendere due libri dalla biblioteca per un esame, a Luigi servono gli stessi due libri, io prendo il libro A, Luigi prende il libro B, ora io dovrei prendere il libro B, ma l'ha già preso Luigi, ok, aspetto che lo renda, intanto mi tengo questo. Ora a Luigi serve il libro A, ma l'ho già preso io, ok, aspetta che io lo renda, intanto si tiene quello. Bocciamo entrambi.

In un computer spesso si vuole dare accesso esclusivo a una risorsa, magari è un file o un handle per non so cosa, e non voglio race condition quindi ci accede solo un processo alla volta, magari è il permesso di usare la stampante e non vorrei avere due processi iniseme che mandano roba alla stampante, che è già stupida di suo. Queste situazioni però possono dare il via all'effetto Luigi illustrato di sopra, come si capisce dal nome della sezione il termine tecnico per l'effetto Luigi è **DEADLOCK**, che suona anche più figo

Il deadlock tra due processi succede quando io sto aspettando il libro di Luigi, ma Luigi sta aspettando il mio libro, visto che ambo i tizii in deadlock stanno aspettando caio, che è in deadlock, non se ne esce.

Come se non bastasse i deadlock possono accadere anche con più di due processi, se c'ho una combriccola di tizii, caii, e sempronii, che stanno tutti aspettando un altro tizio, caio, o sempronio in deadlock, non ne esco ugualmente.

In pseudo-matematiche, un gruppo di processi è in deadlock se ogniuno di quei processi sta aspettando un processo in deadlock

Dato lo pseudo-matematiche non stupirà che abbiamo matematiche dato il matematiche non stupirà che qualcuno si è messo a fare teoremi su sta roba, ahime.

Queste sono delle condizioni necessarie al potersi ritrovare con un deadlock

- una risorsa o ce l'ha un solo tizio o è pigliabile, che ce l'abbia io o che ce l'abbia un qualche processo sono quindi condizioni che si escludono a vicenda, c'è una **mutua esclusione**, facile sennò la dai dai due tizii che la vogliono e non sto più aspettando st'altro stronzo, la definizione casca.
- posso tenermi le risorse che ho acquisito mentre aspetto che mi arrivino le altre risorse che mi servono, se usassi e rilasciassi una risorsa alla volta quella che ho prima o poi la rilascio indipendentemente da che

fanno gli altri, che non sto aspettando niente, poi chiunque mi stia aspettando ce l'avrà poi, non sembra proprio un deadlock.

- niente prelazione di risorse, non possono venire a fottermi il libro per darlo a Luigi, altrimenti il sistema potrebbe sbarazzarsi molto tranquillamente dei deadlock.
- c'è un girotondo chiuso di gente che sta aspettando quello davanti nel girotondo, se non ci fossero girotondi ci sarebbe quello in cima alla coda che non sta aspettando nessuno e si leva, poi quello che era dietro ora è in cima e si leva pure lui, poi si levano tutti, non ha molto l'aria da deadlock.

prendiamo l'ultima condizione, in pseudo-matematiche possiamo dire che

c'è un qualche ciclo nel grafo con i processi come nodi e con $u \rightarrow v$ se u gradirebbe che v si desse una mossa