I processi

Definizione

- Con il termine **processo** si indica una sequenza di azioni che il processore esegue
- Il **programma** invece, è una sequenza di azioni che il processore dovrà eseguire
- Il processo è quindi un programma in via di esecuzione e quindi, dinamico; il programma e i dati, rappresentano le risorse e sono la parte statica

Stati di un processo

- **Pronto**: il processo è stato caricato in memoria
- **Esecuzione**: al processo viene assegnata la risorsa processore per l'esecuzione
- **Attesa**: se si sospende un processo, viene messo in fase di attesa che terminino gli altri
- **Terminazione**: il processore ha terminato l'esecuzione e quindi, il processo deve rilasciare le risorse assegnate

Scheduling

- I S.O. multitasking sono provvisti di moduli detti **scheduler** che assegnano ai processi un ordine, un tempo di esecuzione.
- I processi caricati in memoria non devono però superare un certo limite
- L'azione di selezione degli scheduler è detto **scheduling**
- Esistono **scheduler a lungo termine.** Intervengono nel momento in cui i processi attivati sono in numero superiori. I processi allora vengono caricati nella memoria di massa e, gli scheduler a lungo termine selezionano quelli da caricare in memoria centrale per poi passare nella fase di esecuzione
- Gli scheduler a lungo termine lavorano con una frequenza molto più bassa rispetto a quelli a breve termine
- Gli scheduler a lungo termine devono selezionare in maniera opportuna i processi in modo tale che la coda dei pronti non sia troppo vuota o la CPU troppo impegnata

Interruzioni

Un processo in fase di esecuzione può essere interrotto.

Ogni volta che si genera una interruzione il processo viene posto in fase di attesa finchè viene terminata l'operazione richiesta dopo di che torna nella coda dei pronti. Fa eccezione il caso in cui il processo si interrompe perché ha portato a termine l'esecuzione e quindi, provvede al rilascio delle risorse

Interruzioni

- Si esaurisce il tempo assegnatogli
- Il processo in esecuzione genera una richiesta di I/O
- Il processo genera uno o più sottoprocessi detti figli
- Il processo riceve una interruzione; rilascia il processore e viene posto in attesa fino a che viene portato a termine il servizio invocato dall'interruzione

Tabella dei processi

- I processi sono gestiti tramite una **tabella dei processi** che contiene per ogni processo, il **blocco di controllo del processo**
- Ogni blocco contiene le seguenti informazioni:
  - Lo stato del processo
  - Program counter che contiene l'indirizzo dell'istruzione successiva
  - Il contenuto della CPU
  - Le informazioni dello scheduling che possono essere utili per il context switch
  - Le informazioni relative al tempo di utilizzo della CPU e alla percentuale di CPU impegnata nel processo
  - Allocazione di memoria
  - I/O e i file impegnati nel processo
- Per context swith, si intende il cambio di contesto quando c'è una interruzione. Il S.O deve salvare le informazioni relative al processo in corso per poterlo ripristinare.

Algoritmi di scheduling

I metodi utilizzati per lo scheduling sono i seguenti:

- Round-robin
- Schedulling a code multiple
- Scheduling a code multiple con retroazione
- Scheduling per multiprocessori

Round-robin

- Questo sistema utilizza il metodo della **ripartizione di tempo, time sbaring**; consiste nel formare una coda di processi con accesso FIFO (First Input First Output) e nell'assegnare a ciascuno di essi un tempo che va dai 10 ai 100 ms.
- Durante l'esecuzione di un processo, può capitare che uno di essi venga interrotto e quindi, passa alla coda dei pronti. Per questo motivo, il r-r è detto **scheduling circolare**

Scheduling a code multiple

- I vari processi possono essere posti in code differenti.
- Ad ogni coda viene assegnata una priorità
- Il più delle volte, alla coda sistema, viene data una priorità superiore a quella utente
- I processi di priorità più bassa vengono eseguiti solo quando si è esaurita la coda di quelli a priorità maggiore
- Per ciascuna coda è possibile stabilire i quanti di tempo e la percentuale di utilizzo della CPU

Scheduling per multiprocessori

- Se i processori sono multipli, si può attribuire a ciascuno quali processi eseguire.
- Si parla di multielaborazione asimmetrica, se lo scheduling è assegnato ad un solo processore e agli altri è dato solo il compito di eseguire i processi; si parla di multielaborazione simmetrica se ogni processore seleziona e preleva i processi da una coda comune.
- È opportuno che il processo, una volta caricato su un processore prosegua sempre su quello altrimenti, si può avere un inutile svuotamento e riempimento della memoria cache. Il più delle volte, i processi hanno una predilizione per il processore in cui si trovano così, si ha un aumento delle prestazioni.
- Può capitare che si abbia uno sbilanciamento del carico, cioè, alcuni processori sono sovraccarichi e altri fermi perché privi di processi in corso. Si procede allora ad un bilanciamento del carico tramite una migrazione che può essere o spontanea,o guidata da altri processi

Scheduling per multiprocessori: iperthread

- Le considerazioni fatte per un s.o multiprocessore può essere esteso anche ad alcuni s.o. che vedono i processori come un insieme di processori logici che provvedono alla tecnica Iperthread. Molti di questi processori sono della Intel.
- Il sistema Windows XP utilizza un metodo di prelazione con indici che vanno da 0 a 31. La massima priorità è 31.
- Per ogni priorità viene creata una coda di pronti; il modulo del kernel gestisce la schedulazione ed è detto dispatcer.
- Il dispatcher è quindi, un modulo del kernel che scandisce ogni coda da quella con maggiore priorità a quella con più bassa priorità alla ricerca di thread pronto.

Comunicazioni tra processi

- Due processi si dicono indipendenti se non si condizionano
- Se un processo condiziona l'altro allora i due si dicono **cooperanti**
- I processi spesso condividono i dati e quindi, devono comunicare tramite un meccanismo detto **IPC: InterProcess Comunication**
- La comunicazione tra processi può avvenire o per memoria condivisa, se i processi cooperanti accedono ai dati contenuti nell'area in cui si scambiano informazioni, oppure con scambio di messaggi se i processi utilizzano l'invio e la ricezione di messaggi per comunicare.

Comunicazioni tra processi

- Un processo che produce informazioni è detto **produttore**, mentre chi utilizza le informazioni è detto **consumatore**
- Se i dati da scambiare sono di notevoli dimensioni, conviene utilizzare un certo spazio di memoria condivisa, altrimenti, si possono utilizzare dei semplici messaggi con funzioni primitive, **send** e **receive**
- si chiama comunicazione diretta, quella che provvede ad inviare il messaggio dal processo mittente al processo destinatario; si indica con comunicazione indiretta, quella che provvede ad inviare il messaggio dalla porta mittente alla porta destinatario.
- Le porte sono quindi delle interfacce dette **mailbox**
- In Windows lo scambio tra messaggi è detto **LPC,** ossia **Local Procedure Call**; ciò avviene sullo stesso computer e, il processo produttore è detto server mentre il consumatore è detto client. Il client effettua una richiesta di connessione al server aprendo una maniglia detta **handle**; il server apre due porte per la connessione. La memoria condivisa è detta **oggetto di sezione**

Competizioni tra processi

- Due o più processi possono essere in competizione tra loro se accedono alle stesse risorse che spesso sono limitate
- Si definisce quindi **corsa critica**, la competizione tra processi.
- La soluzione ai problemi è il principio della **mutua esclusione** che impedisce agli altri processi di poter accedere alle risorse in uso da un solo e determinato processo
- La parte di codice con cui il processo accede alle risorse condivise, si chiama **sezione critica**

Sincronizzazione tra i processi

- Una valida alternativa alle competizioni tra processi è la sincronizzazione
- I semafori sono delle variabili che gestiscono la sincronizzazione
- A queste variabili denominate S, si accede tramite due operazioni predefinite: wait e signal