

UF2. Programació de processos i fils

Processos

- Concurrència -

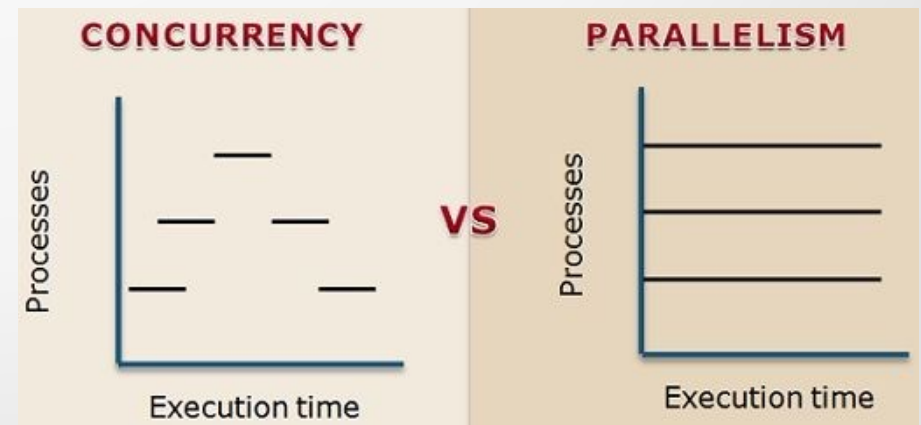
Processos vs Serveis

- Els processos que no tenen interfície gràfica i s'executen en segon pla (*background*) són coneguts en Windows com **serveis** o en Linux com a **daemon** (*Disk And Execution Monitor*, popularment, dimonis).
- Poden estar en execució local com ara el Firewall, antivirus, la connexió Wi-Fi... o bé en un sistema distribuït o Internet com el servei d'HTTP, DNS, FTP, etc.



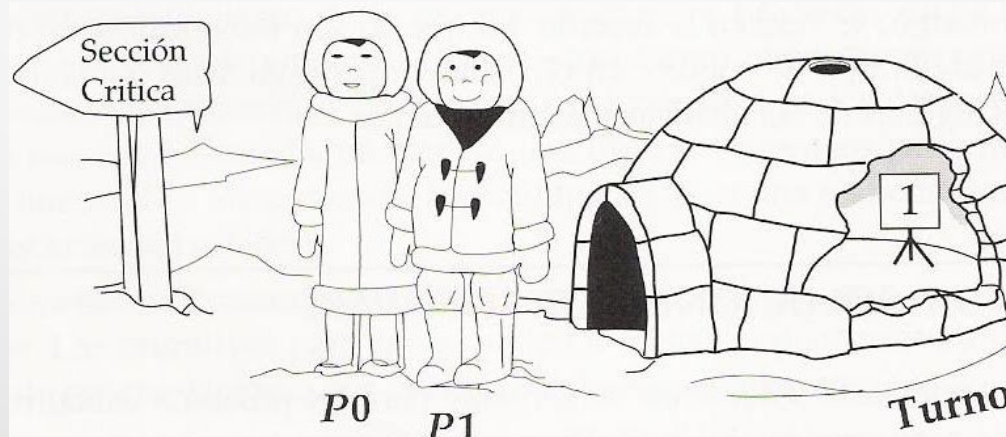
Sincronització de processos

- En la gestió de processos, la **concurrència** és la tasca més complexa. El seu objectiu és garantir la sincronització de l'execució dels diferents processos perquè el resultat final sigui l'esperat, aprofitant el màxim el temps del processador.
- Els **problemes** que es plantegen són:
 - l'accés als recursos compartits per part de dos processos.
 - assignacions de recursos



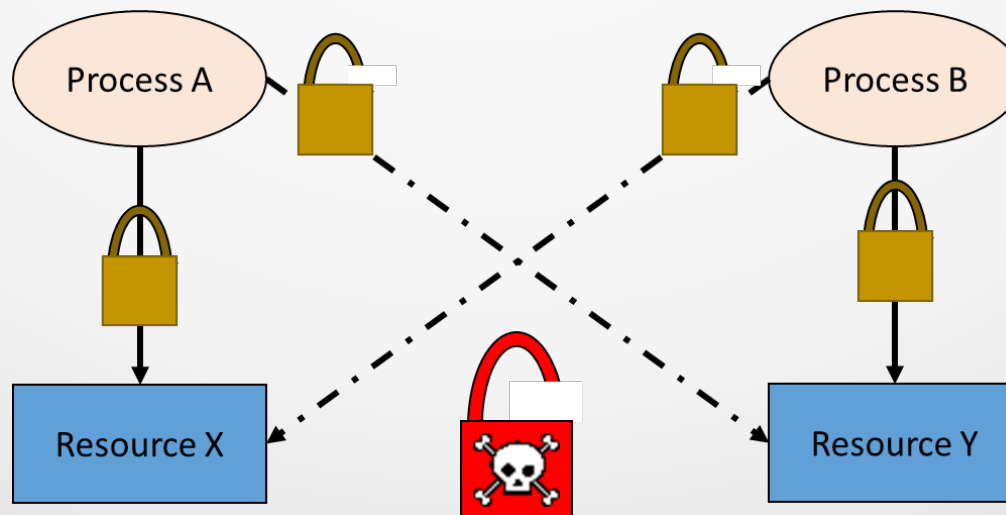
Sincronització de processos

- En cas que els processos mostrin competència per algun recurs, el S.O haurà de ser capaç d'abordar 3 problemes:
- **Exclusió mútua** (*MUTEX, mutual exclusion*): la part dels processos on se'n fa ús d'aquest recurs no compartible passarà a dir-se **secció crítica** i solament un procés podrà entrar en la seva secció crítica en un moment donat. S'elimina la possibilitat que els dos accedeixin al recurs a la vegada, però es creen dos nous problemes:



Sincronització de processos

- **Interbloqueig** (*deadlock*): es pot produir en el cas que dos processos necessitin a dos recursos per poder entrar a la seva secció crítica i el S.O els hi atorgui un a cada un. Els dos processos es quedarien bloquejats indefinidament.
- **Inanició**: un procés no se li arriba a assignar mai un recurs pel que no pot entrar en la seva secció crítica.



Sincronització de processos

- Al llarg de la història, s'ha proposat solucions específiques per solucionar els problemes de sincronització: semàfors, monitors i pas de missatges.
 - **Semàfors**: variables especials per a restringir o permetre l'accés a recursos compartits en funció si es troba bloquejat (vermell) o alliberat (verd).
 - Existeixen dos tipus de semàfors:
 - **Semàfor binari**: Consta de dues operacions `Signal(S)` i `Wait(S)` que s'executen com a úniques accions atòmiques.

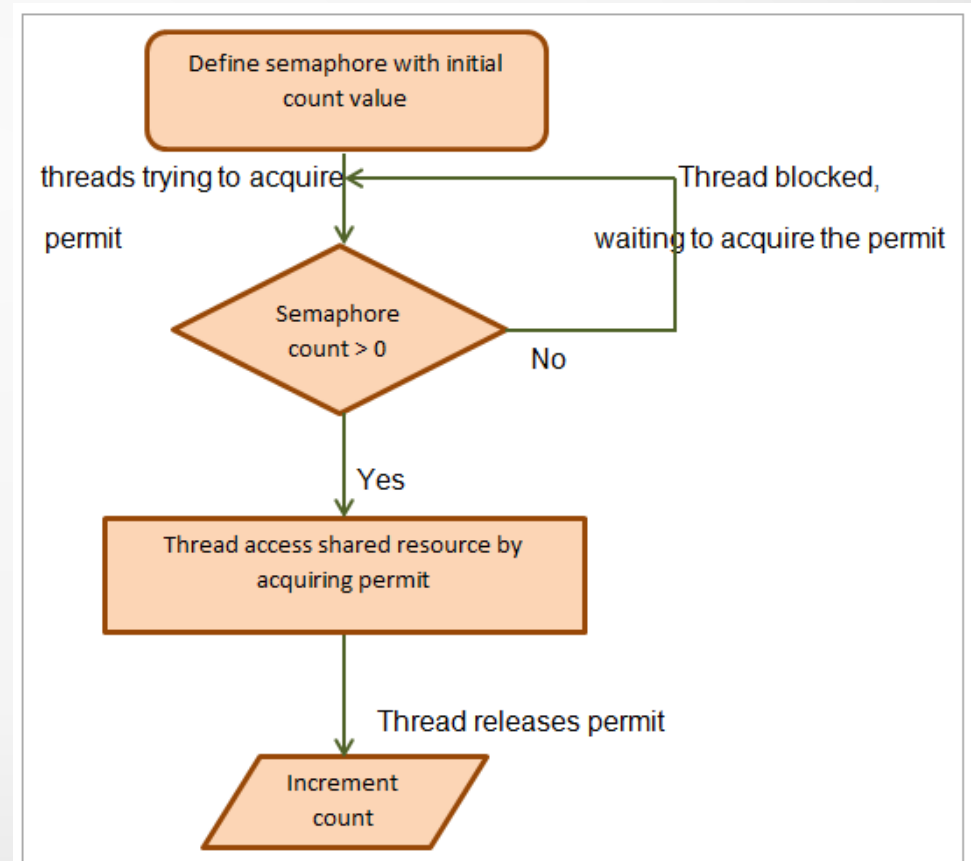


Sincronització de processos

- **Semàfor amb comptador:** tenen n permisos que atorguen al processos que volen fer ús del recurs compartit. Operacions:

- **acquire():** el procés sol·licita un permís.

- **release():** el procés retorna el permís.



Sincronització de processos

- **Monitors:** algoritme que fa una abstracció de dades que permet representar un recurs compartit mitjançant un conjunt de variables que defineixen el seu estat. L'accés és possible des d'uns mètodes del monitor.

Similar a una habitació amb una sola porta i els recursos compartits dins. Només un procés pot entrar dins, complint les condicions del monitor.

- **Pas de missatges:** paradigma on emissor envia missatge a receptor, podent fer ús d'una bústia (adreçament indirecte).

