

Scheduling:

Round Robin: algo d'ordonnancement
IO/ CPU BOUND

2 types de process :

- calculatoire, run -> ready, consomme le quantum
=> CPU Bound
ex: une machine plus puissante => plus rapide
- Interactif, run -> waiting, ne consomme le quantum
=> IO Bound

Programme interactif, sleep jusqu'à l'arrivée d'un event

Programme calculatoire, ça fait quelque chose et hop c'est fini (ex: un compilateur)

On cherche un algo qui va prendre en compte le type de process présent, ce que Round Robin ne fait pas

Waiting time: le temps que le process a été en ready avant d'être appelé

Comment savoir si process le CPU ou IO Bound ?

- On peut regarder le temps en running
- Le nombre de fois en state wait

LRU: Least Recently Used

Aging: on va maintenir un entier qu'on va shift (bit de poids fort) à chaque fois que le process est utilisé.

Swap: si j'ai plus de mémoire principale (RAM), on décharge des trucs sur le disque

Les utilisations mémoires:

on a deux bits : access et dirty qui sont mis à 1,

Le cache TLB: mémoire cache du processeur utilisé par l'unité de gestion mémoire dans le but d'accélérer la traduction des adresses virtuelles en adresses physiques.

File System

Inode (faut voir le schéma)

Inode sur le disque : format du fichier

Inode en mémoire : sur l'OS

mount

Threads

Communication entre deux process:

- pipe
- named pipe/ fifo, mkfifo
- socket
- unix socket (on peut envoyer un fd)
local socket
utilisé pour faire de la séparation de privilège

on en a un qui lit et l'autre qui écrit