

Systeme d'Exploitation

Mohamed TOUNSI

Institut Supérieur d'Informatique et de Multimédia Sfax

Février 2015

Introduction aux processus

Définitions (1)

Définition d'un processus

Un processus est un programme en cours d'exécution. Il possède son propre compteur ordinal, ses registres et ses variables.

- Le processus est créé par le système d'exploitation ou l'utilisateur au moment où l'exécution du programme doit commencer,
- Une fois le processus terminé, il est supprimé par le système d'exploitation,
- Toutefois, un seul programme peut nécessiter plusieurs processus pour son exécution.

Introduction aux processus

Définitions (2)

Les principales fonctionnalités du système d'exploitation en matière de gestion de processus consistent à :

- La création, suppression et interruption de processus,
- L'ordonnancement des processus afin de décider d'un ordre d'exécution équitable entre les utilisateurs tout en privilégiant les processus du système,
- La synchronisation entre les processus ainsi que la communication,
- La gestion des conflits d'accès aux ressources partagées,
- La protection des processus d'un utilisateur contre les actions d'un autre utilisateur.

Introduction aux processus

États d'un processus (1)

Un processus passe par plusieurs états avant de finir son exécution:

- **prêt**: le processus est prêt et pourra s'exécuter dès qu'il sera sélectionné par le système d'exploitation,
- **En exécution**: le processus est en cours d'exécution au niveau du processeur,
- **En attente**: le processus est en attente ou bloqué car il attend des données ou la libération d'une ressource afin de poursuivre son exécution,
- **Terminé**: le processus a terminé son exécution.

Introduction aux processus

États d'un processus (2)

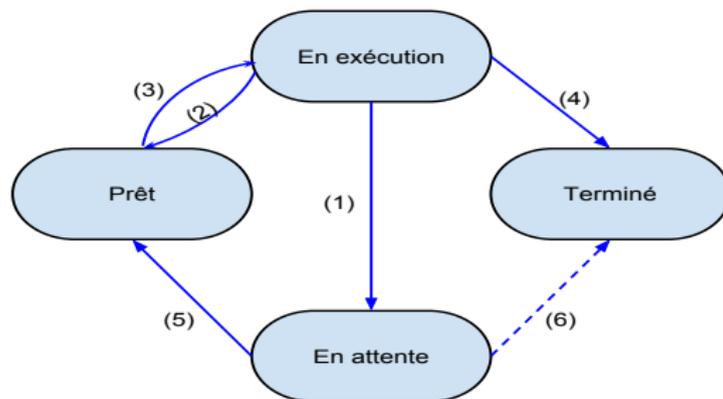


Figure : Cycle de vie d'un processus

Introduction aux processus

Structure de l'espace mémoire d'un processus

L'espace mémoire utilisé par un processus est divisé en plusieurs zones:

- **Données Statiques:** les variables globales ou statiques, ainsi que les allocations dynamiques,
- **Instructions:** code des instructions du programme à exécuter,
- **Pile:** sert à empiler les appels de fonctions, avec leurs paramètres et leurs variables locales.



Figure : Le processus dans la RAM

Interruption d'un processus

Fonctionnement d'une interruption

- Une interruption est provoquée par un signal généré sur occurrence d'un événement qui peut être interne (lié au processus) ou externe et indépendant,
- Lorsqu'une interruption est générée, le processus en cours d'exécution est interrompu,
- Il quitte le processeur et un gestionnaire d'interruption est chargé dans les registres du processeur et s'exécute pour traiter l'interruption.

Interruption d'un processus

Types d'interruptions d'un processus

Un processus peut être interrompu :

- interruptions externes:
 - Panne,
 - Intervention de l'utilisateur,...
- Interruptions internes:
 - Division par zéro, débordement de la mémoire, exécution d'une instruction non autorisée,...
 - Quand une instruction effectue un appel système.

On distingue alors deux sortes d'interruptions: **logicielles** et **matérielles**.

Structures de données pour la gestion des processus

Contexte d'un processus (1)

Les systèmes d'exploitation manipulent deux structures de données principales pour gérer les processus créés sur une machine :

- 1 **La table des processus** et
- 2 **Le bloc de contexte d'un processus.**

Définition

- Le contexte d'un processus (appelé aussi bloc de contexte ou de contrôle) est une structure de données qui décrit un processus en cours d'exécution,
- Il est créé au même moment que le processus,
- Il est mis à jour en grande partie lors de l'interruption du processus afin de pouvoir reprendre l'exécution du processus ultérieurement.

Structures de données pour la gestion des processus

Contexte d'un processus (2)

Le contexte d'un processus comporte les informations suivantes:

- Le compteur ordinal: l'adresse de la prochaine instruction à exécuter par le processeur,
- Les contenus des registres généraux: ils contiennent les résultats calculés par le processus,
- Les registres qui décrivent l'espace qu'il occupe en mémoire centrale (l'adresse de début et de fin par exemple),
- Le registre variable d'état qui indique l'état du processus
- D'autres informations telles que la valeur de l'horloge, la priorité du processus,

Structures de données pour la gestion des processus

La table des processus

Définition

La table des processus contient toutes les informations indispensables au système d'exploitation pour assurer une gestion cohérente des processus.

- Elle est stockée dans l'espace mémoire du système d'exploitation, ce qui signifie que les processus ne peuvent pas y accéder.
- Elle comporte une entrée par processus:
 - des informations concernant un processus (même si le processus n'est pas en mémoire),
 - des informations sur les fichiers qu'il manipule,
 - des informations sur son occupation mémoire (des pointeurs sur les différents segments code, données et pile.)...

Ordonnancement de processus

Pourquoi ?

Le maintien du taux d'utilisation du processeur est représenté par différentes valeurs:

- Le taux d'utilisation de l'unité centrale,
- Le débit,
- Le temps de traitement moyen,
- Le temps de traitement total,
- Le temps de réponse maximal.
- etc..

Objectif

L'ordonnancement des processus permet essentiellement aux processus de s'exécuter à un moment ou un autre et d'utiliser au mieux le processeur pour l'utilisateur.

Ordonnement de processus

Présentation

définition

L'ordonnement de processus s'articule autour de 2 composantes:

- 1 **La file d'attente des processus:** elle regroupe les descripteurs des processus prêts. C'est à partir de cette file que l'ordonneur choisit le processus à exécuter,
- 2 **L'algorithme d'ordonnement** à appliquer: il organise les descripteurs de processus dans un ordre qui reflète la stratégie appliquée.

Ordonnancement de processus

L'ordonnanceur

- Le système d'exploitation applique un programme pour gérer l'ordre d'exécution des processus, appelé scheduler, dispatcher ou ordonnanceur,
- L'ordonnanceur dispose de plusieurs algorithmes d'ordonnancement qui correspondent à différentes politiques d'organisation des processus: équitable, avec priorité, plus court d'abord,
- Il agit sur la file des processus prêts et l'organise en appliquant un algorithme donné,
- L'ordonnanceur ne change pas de politique d'ordonnancement,
- Chaque système d'exploitation applique un algorithme choisi par les concepteurs.

Ordonnancement de processus

Algorithmes d'ordonnancement

Soient les processus suivants décrits par une durée d'exécution estimée en millisecondes.

Processus	Durée d'exécution
P1	3
P2	5
P3	2
P4	6
P5	4

Tab.1 Processus et leurs temps d'exécution

Pour chaque algorithme (voir dessous), nous calculons:

- Le **temps de réponse** est la moyenne des dates de fin d'exécution.
- Le **temps d'attente** est la moyenne des délais d'attente pour commencer une exécution.

Ordonnement de processus

L'ordonnement FIFO: First In First Out (1)

Principe

Le premier processus arrivé est le premier servi. Il n'y a aucun effort à faire de la part de l'ordonnanceur, il suffit d'insérer le processus à la fin de la file.

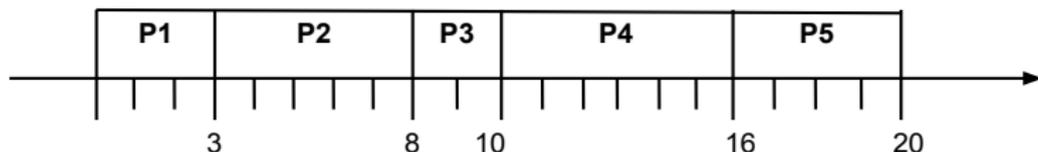


Figure : Ordonnement des 5 processus de la Tab.1 avec la méthode FIFO

Ordonnancement de processus

L'ordonnancement FIFO: First In First Out (2)

- Temps de réponse $TR = (3+8+10+16+20)/5 = 57/5 = 11.4$ ms,
- Temps d'attente $TA = (3+8+10+16)/5 = 37/5 = 7.4$ ms

Conclusion

- Appliquer la politique FIFO revient à faire de la monoprogrammation en quelque sorte,
- Cet algorithme est souvent retenu du fait de sa simplicité et de sa faible consommation en temps processeur.

Ordonnancement de processus

L'ordonnancement circulaire : le tourniquet (Round Robin) (1)

Principe

- 1 Le premier processus de la file est exécuté durant une période précise fixée par le système, appelée quantum,
- 2 Une fois le processeur est alloué, il s'exécute jusqu'à ce que sa tranche de temps soit écoulee ou qu'il se bloque,
- 3 Une fois interrompu, il sera mis à la fin de la file d'attente (fonctionnement circulaire),
- 4 L'ordonnanceur retire le processus suivant et l'exécute.

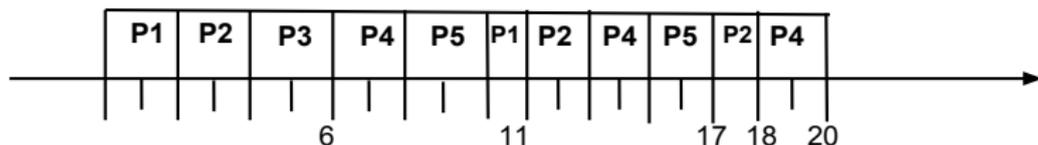


Figure : Ordonnancement des 5 processus de la Tab.1 avec la méthode tourniquet

Ordonnancement de processus

L'ordonnancement circulaire : le tourniquet (Round Robin) (2)

- Temps de réponse $TR = (6+11+17+18+20)/5 = 72/5 = 14.4$ ms
- Temps d'attente $TA = ((8)+(2+7+4)+(4)+(6+5+3)+(8+5))/5 = 52/5 = 10.4$ ms

Remarque

- Le choix de la durée d'exécution (le quantum), doit être fait avec précaution.
- Si ce temps est faible et comparable à celui que prend le changement de contexte, le processeur se retrouve à passer plus de temps à charger et décharger des processus plutôt qu'à les exécuter.

Ordonnement de processus

L'ordonnement selon le plus court d'abord (SJF Shortest Job First) (1)

Principe

Comparer les processus sur la base de leur durée d'exécution et exécuter le plus court en premier.

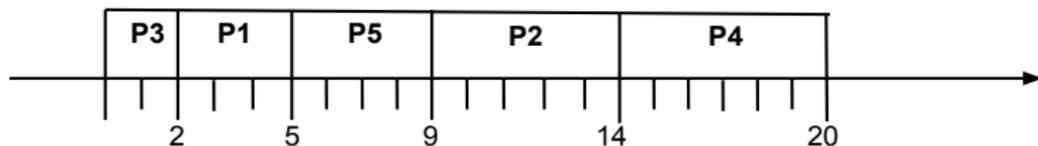


Figure : Ordonnement des 5 processus de la Tab.1 avec la méthode SJF

Ordonnancement de processus

L'ordonnancement selon le plus court d'abord (SJF Shortest Job First) (2)

- Temps de réponse $TR = (2+5+9+14+20)/5 = 50/5 = 10$ millisecondes
- Temps d'attente $TA = (2+5+9+14)/5 = 30/5 = 6$ millisecondes

Remarque

Le SJF est appliqué dans les systèmes qui régissent les processus industriels où une chaîne d'exécution doit se terminer le plus rapidement possible.

Ordonnement de processus

L'ordonnement avec priorités (1)

Principe

- Distinguer entre les processus sur la base de leurs priorités,
- L'utilisateur et le système d'exploitation peuvent attribuer les priorités,
- L'algorithme d'ordonnement avec priorité classe les processus avec un ordre décroissant de leur priorité et le processus ayant la plus haute priorité est exécuté en premier.

Si les processus ci-dessus présentés avaient respectivement les priorités 1, 3, 2, 5 et 4, ils seraient exécutés selon l'ordre suivant:

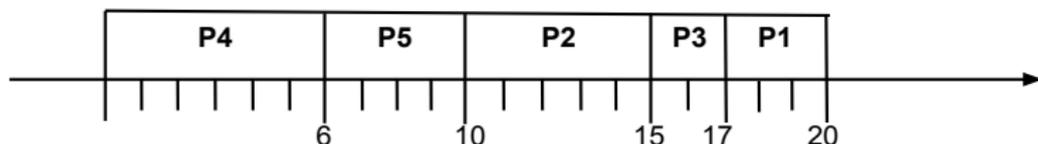


Figure : Ordonnement des 5 processus de la Tab.1 basé sur la priorité

Ordonnancement de processus

L'ordonnancement avec priorités (2)

- Temps de réponse $TR = (6+10+15+17+20)/5 = 68/5 = 13.6$ ms
- Temps d'attente $TA = (6+10+15+17)/5 = 48/5 = 9.6$ ms

Remarque

Chaque processus s'exécute jusqu'à la fin sinon la priorité n'exprime aucune urgence. A signaler que les processus du système d'exploitation ont la priorité la plus haute.

Problème de la famine

Le problème de la famine peut se poser s'il existe plusieurs processus de haute priorité. Ils monopoliseront l'unité centrale et la file des processus de faible priorité ne sera servie que très rarement.

Ordonnancement de processus

Résolution du problème de la famine

- Parmi les solutions qui existent, la plupart décrémentent la priorité du processus où après exécution d'un quantum ou à chaque impulsion d'horloge.
- Ainsi à un moment donné, sa priorité deviendra inférieure à celle d'un autre processus qui prendra alors la main évitant ainsi la famine.
- On distingue plusieurs autres approches pour l'ordonnancement comme celui basé sur l'exécution du plus court processus d'abord, ou bien l'ordonnancement avec ou sans préemption....

Sources

- Au cœur des systèmes d'exploitation des ordinateurs: Concepts de base et exercices résolus, Leila Baccouche, Tunis, 2003.
- Systèmes d'exploitation: Cours et exercices corrigés, J. Archer Harris, 2002.