

# 

# Programmation système :

# PROSIT 6 ALLER :

# Synchronisation

# A3

# Analyse du contexte

Mathilde et Frank ont trouvé une solution pour lancer plusieurs tâches de processus en parallèle mais elles ne s’appliquent pas à tous les niveaux.

# Définition mots-clefs

Simulateur : Un simulateur est un programme informatique ou un dispositif matériel qui imite le comportement d'un système réel. Il est utilisé pour tester, entraîner ou comprendre le fonctionnement d'un processus, d'un appareil ou d'un système complexe.

Synchronisation : La synchronisation dans le contexte informatique fait référence à la coordination et à la gestion ordonnée des actions entre différents processus, threads ou composants pour éviter des résultats inattendus ou des conditions de concurrence.

Stratégie de communication inter-processus : Une stratégie de communication inter-processus définit la manière dont les processus (programmes en cours d'exécution) échangent des données et communiquent entre eux. Cela peut inclure l'utilisation de files d'attente, de signaux, de mémoire partagée, etc.

Multi-Threading : La multi-threading est une technique où plusieurs fils d'exécution (threads) partagent les ressources d'un même processus. Cela permet l'exécution simultanée de plusieurs tâches, améliorant ainsi l'efficacité d'un programme.

Anonymous Pipe : Une anonymous pipe (tube anonyme) est un mécanisme de communication inter-processus dans lequel un processus peut écrire des données dans un tube, et un autre processus peut les lire. Il est appelé "anonyme" car il ne nécessite pas une identification explicite.

Inter-blocage : L'interblocage (ou deadlock) est une situation dans laquelle deux ou plusieurs processus ou threads ne peuvent pas progresser en raison d'une attente mutuelle de ressources. Cela peut entraîner une paralysie du système.

Mécanisme d'IPC (Inter-Process Communication) : Les mécanismes d'IPC sont des moyens par lesquels les processus peuvent communiquer et partager des données entre eux. Cela peut inclure des pipes, des files d'attente, des signaux, des sockets, etc.

Section/ressource critique : Une section critique (ou ressource critique) est une partie d'un programme où l'accès concurrent de plusieurs threads ou processus à une ressource partagée doit être coordonné pour éviter des incohérences ou des conflits.

# Problématique

**- Comment implémenter le mécanisme de synchronisation fonctionnel ?**

# Contraintes

Schéma de la chaîne de production

# Généralisation

Effectuer un mécanisme de synchronisation.

# Livrables

Code fonctionnel simulant la chaîne de production.

Stratégie de synchronisation : diagramme de séquence

# Pistes de solution

Reprendre les objets créés au Prosit 2

Utiliser WaitForExit()/Thread.Sleep()

Reprendre la méthode de pseudo-simultanéité du prosit 5

Utiliser les mécanismes de synchronisation Mutex et Sémaphore

Ordonner chaine de production (en parallèle) en commençant par les machines les plus longues

# Plan d’action

Etudier stratégie de communication (IPC)x

Faire diagramme de séquence

Appliquer la pseudo-simultanéité sur la chaine de production

Implémenter la synchronisation pour répondre à la problématique

# Réalisation du plan d’action