

---

# AI01

## TP3

---

REDOUIN INNECCO Gabriel  
GAMONDELE Maxime  
29 octobre 2025

Ce document synthétise le travail et la réflexion sur le TD de XXX

...

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>First Come, First Served</b>	<b>2</b>
2.1	Introduction . . . . .	2
2.2	Résultats et analyse . . . . .	2
<b>3</b>	<b>Shortest Job First</b>	<b>3</b>
3.1	Introduction . . . . .	3
3.2	Résultats et analyse . . . . .	3
<b>4</b>	<b>Conclusion</b>	<b>4</b>



# 1 Introduction

L'ordonnancement des processus est une composante essentielle des systèmes d'exploitation, car il détermine la manière dont les processus se partagent le processeur. Selon la politique choisie, l'objectif peut être de maximiser l'efficacité, d'assurer une certaine équité ou encore de réduire le temps d'attente des tâches.

Les listes chaînées sont des structures qui permettent de stocker et de manipuler facilement des processus, que ce soit sous la forme d'une file FIFO, d'une pile LIFO, d'une liste triée, ou encore d'autres types de structures.

La première politique étudiée est le FCFS (First Come, First Served), où les processus sont exécutés dans l'ordre d'arrivée. Pour la simuler, une file FIFO est utilisée, permettant d'ajouter les processus à la fin de la file et de les retirer au début. La seconde politique est le SJF (Shortest Job First), où les processus les plus courts sont traités en priorité. Dans ce cas, une liste triée est utilisée afin d'insérer chaque processus à la position correcte, garantissant que les durées restent classées dans l'ordre croissant.

L'objectif de ce TP est ainsi de comprendre le fonctionnement et la manipulation des listes chaînées tout en observant l'impact concret de chaque politique sur l'exécution des processus.

## 2 First Come, First Served

### 2.1 Introduction

L'ordonnancement FCFS (First Come, First Served) exécute les processus selon leur ordre d'arrivée. Ceux-ci sont stockés dans une file d'attente de type FIFO (First In, First Out), où le premier processus inséré est également le premier retiré. L'implémentation utilise cette structure pour gérer simplement la succession des exécutions. Chaque processus terminé libère la place au suivant dans la file. Ci dessous un schéma représentant la file FIFO (cf. Figure 1).

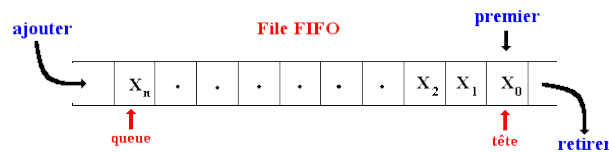


FIGURE 1 – Structure de la file FIFO pour FCFS

### 2.2 Résultats et analyse

Après avoir implémenté la politique FCFS à l'aide d'une file FIFO, nous avons observé les résultats suivants : De plus, nous avons analysé les performances de cette politique en termes de complexité. Ici pour notre algorithme FCFS, la complexité est de  $O(n)$  car chaque processus doit être ajouté et retiré de la file une seule fois.



## 3 Shortest Job First

### 3.1 Introduction

L'ordonnancement SJF (Shortest Job First) exécute en priorité les processus dont la durée d'exécution est la plus courte. Les processus sont organisés dans une liste triée en fonction de leur temps d'exécution, ce qui permet de sélectionner en premier celui dont la charge est minimale. L'implémentation repose sur cette structure ordonnée qui facilite la recherche du processus le plus court à chaque étape. Une fois un processus terminé, le suivant le plus court est choisi automatiquement. Ci-dessous (cf. Figure 2) un schéma illustrant la liste triée utilisée pour cet ordonnancement.

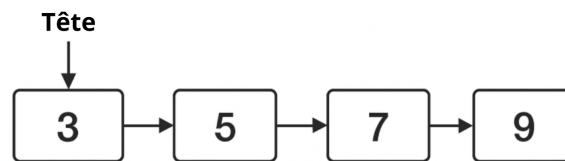


FIGURE 2 – Structure de la file FIFO pour FCFS

### 3.2 Résultats et analyse



## 4 Conclusion

Dans ce TP, nous avons