BONTEMS Antoine 2TL2

WU Jiale 2TM1

SCHOONYANS Ann-Lore 2TL2



Rapport du Projet de Système d’exploitation II :

Championnat de Formule 1

Année 2024 - 2025

**Introduction**

Le projet permet de gérer les Grands prix d’un championnat de Formule 1 en C. Le programme couvre les différentes étapes de tous les week-ends sur l’année, depuis les essais libres jusqu’à la course finale du dimanche, incluant la course sprint pour les week-ends spéciaux.

Chaque course génère des temps aléatoires par secteur, gère la gestion des arrêts aux stands, et le maintien d’un classement dynamique basé sur les performances des pilotes. Les données des différentes courses doivent être sauvegardées pour la réalisation du Grand Prix en son entièreté.

Ce projet permettra également de montrer la synchronisation des processus via des sémaphores pour gérer les accès à la mémoire partagée.

**Prérequis**

Afin de garantir le bon fonctionnement du programme, les fichiers pour les qualifications (q1.txt, q2.txt, q3.txt) doivent être présents. Le fichier q1.txt doit être initialisé avec les numéros des pilotes participants séparés par un espace (ex : 1 11 44 63 16 55 4 81 14 18 10 31 23 2 22 3 77 24 20 27). Pour avoir un affichage correct, le premier Grand Prix doit être réalisé dans le bon ordre.

**Analyse**

Le code permet de choisir quelle étape du Grand Prix on souhaite lancer ainsi que le nombre de tours ou le temps en minute, si les étapes ne sont pas en ordre chronologique, on se base sur les derniers résultats sauvegardés de « l’étape » précédente (potentiellement ceux du Grand Prix précédent) dans les fichiers qui doivent donc être présents au préalable.

Le programme permet de créer 20 processus qui simuleront des voitures, ces voitures feront lors de chaque tour des temps aléatoires et seront classées en fonction de ceux-ci. Elles peuvent également finir out ou au pit lors de chaque tour, ce qui peut impacter leur temps.

Ces 20 processus stockeront ensuite leurs temps dans une mémoire partagée qui sera copié puis traitée par le processus parent pour en faire le tri et l’affichage en console. Les pilotes ayant fait les meilleurs temps par secteur et le meilleur tour sont également affichés.

Pour que le parent puisse faire cette copie, un algorithme de Courtois inversé permet d’empêcher toute lecture lorsqu’une écriture est en cours par l’un des 20 processus voiture, ce qui permet de s’assurer qu’ils aient tous terminé d’écrire avant de lire.

Une fois l’étape terminée, les résultats sont stockés dans des fichiers qui permettent de gérer les étapes qui suivront en sachant notamment quels pilotes seront qualifiés ou l’ordre de la grille de départ pour la course du dimanche.

**Conclusion**

Toutes les fonctionnalités demandées ont été implémentées, bien que certaines aient nécessité des interprétations créatives. Par exemple, lorsque les étapes ne suivent pas un ordre logique (exemple : Q3 avant Q1), le programme se base sur les résultats sauvegardés précédemment pour assurer un fonctionnement minimal, offrant ainsi une continuité tout en garantissant une expérience utilisateur correcte.

En terme d’amélioration, le nombre de tour pourrait être donné différemment en donnant par exemple le nom du circuit qui donnerait donc un certain nombre de tours en fonction de sa taille au lieu de directement passer le nombre de tour en argument.

Une autre amélioration serait de gérer plus de cas d’erreurs spécifiques en avertissant l’utilisateur lorsqu’elles ont lieu mais également l’informer que si le Q3 est joué en 1er, les résultats du Q2 d’un ancien Grand Prix seront pris en compte.

Une de nos difficultés a été que le temps imparti en classe (1h/semaine) n’était pas suffisamment long pour avancer efficacement étant donné que, le temps de se décider sur une fonctionnalité à implémenter et de se remémorer le code déjà présent, une bonne partie de l’heure était déjà écoulée.

**Note du groupe**

Ce projet a permis d’approfondir nos connaissances en programmation concurrente et en gestion des processus. Bien qu’il ait présenté des défis techniques et organisationnels, il nous a permis de comprendre les concepts abordés aux cours théoriques et de nous apprendre à utiliser le C sous Linux ainsi que de travailler en groupe.

**Lien GitHub**

<https://github.com/ZosiscoIV/Grand_Prix_F1_OS>