

TP3 : Gestion d'un parc de vélos

Groupe : G7

Étudiants : Atheeshan Vigneswaran & Ismaïl Abloua

Statut du projet :

Le projet est entièrement fonctionnel. Aucun bug n'a été détecté lors des tests avec les paramètres recommandés (S=10, H=100, M=10, N=10). Le comportement du programme respecte l'ensemble des contraintes spécifiées, et le nombre total de vélos est vérifié en fin d'exécution. Les attentes sont toutes passives et le programme utilise correctement les mécanismes de synchronisation.

Résumé de l'implémentation :

Chaque habitant est représenté par un thread. Il effectue M trajets entre des sites tirés aléatoirement, en prenant un vélo s'il est disponible et en attendant passivement dans le cas contraire. Une fois arrivé à destination, il tente de rendre le vélo. Si aucune place n'est disponible, il attend passivement également.

Le camion est représenté par un thread séparé. Il parcourt en boucle les sites pour rééquilibrer les vélos. Il retire des vélos aux sites surchargés (plus de N-2 vélos) tant que sa capacité le permet, et en ajoute aux sites sous-chargés (moins de 2 vélos). Il rééquilibre également son contenu avec le dépôt pour toujours transporter 2 vélos si possible. Le camion continue tant que tous les habitants n'ont pas terminé leurs trajets.

Gestion de la concurrence :

Les accès concurrents aux variables critiques sont protégés par des mutex "site\_mutex". Les sémaphores permettent de gérer l'accès aux vélos "sem\_bikes" et aux places de parking "sem\_slots" de façon passive. Toutes les sections critiques sont réduites au minimum pour maximiser le parallélisme.

Le programme a été testé sans les délais "usleep()" pour valider l'absence de blocages, la cohérence des valeurs et la terminaison correcte de tous les threads.