

PROJET PPII - Compte-rendu N°5

<u>Motif / type de chantier :</u> Réunion de chantier	<u>Lieu :</u> S0.qqch + Mathis en ligne
<u>Présents :</u> Mathis MANGOLD Hélène BARBILLON Adrien LAROUSSE Aurélien GINDRE	<u>Date :</u> 12/05/2023 <u>Durée :</u> 13h-14h

Ordre du jour :

- Fork (Mathis)
- Partie simulation (Aurélien)
- Affichage de la simulation (Adrien)
- Tests
- Ce qu'il reste à faire
- Rapport du projet

Résumé de la réunion :

- Fork de Mathis :

Mathis explique le fork = plusieurs programmes qui tournent en même temps, c'est plus rapide que si on lançait un programme plusieurs fois d'affilées, puisque plus de ressources de l'ordinateur sont utilisées.

Aurélien demande si on ne peut pas plutôt allouer plus de ressources au programme pour le lancer plusieurs fois d'affilée : c'est possible mais compliqué et pas l'approche qui a été choisie.

Adrien demande si ça ne pose pas problème d'utiliser 100% de la capacité de l'ordinateur. Mathis veut exploser les ordinateurs.

2ème option : le multithreading = permet de créer aussi des autres processus appelés threads (mais ce n'est pas un processus à part entière qui tourne à côté, ça utilise un peu moins de mémoire).

Il faudra mettre la fonction qui permet de calculer les trajets dans un autre fichier que le main.c sinon on ne peut pas la tester.
- Partie simulation (Aurélien) :

L'implémentation de la file d'attente est presque finie.

Le plus gros soucis que pose le multithreading est que pour la partie simulation, il y a besoin du trajet précédent pour le suivant, on ne peut pas tout faire en parallèle.

Il faut aussi modifier ce qui est renvoyé en sortie pour que ce soit exploitable pour la partie affichage de la simulation. Plusieurs formats pour la sortie ont été évoqués, on retient celui envoyé par Aurélien sur discord.

Aurélien va modifier les indices id_bornes de 0 à 17000 au lieu de 0 à 63000.
- Affichage de la simulation (Adrien) :

Une page web en plus a été ajoutée au site qui nous permet d'afficher les trajets, naviguer entre les ticks ne sera pas un problème si les données sont bien envoyées sur le site.

Est-ce que ce sera l'utilisateur qui choisira le nombre de trajets qu'il lance ? Oui, il faut donc rajouter des entrées utilisateurs.

Il reste également à personnaliser les points affichés sur la carte en fonction de si les bornes sont en attente ou occupées, on ne représentera pas les bornes vides.
- Tests :

Contenu du dossier app : src, data, test

Oster nous a aidé à faire fonctionner le makefile (ce boss)

On teste 2 choses : le temps de calcul de l'algorithme de la partie 1 et la précision de la fonction de calcul de distance (distance du trajet trouvée comparée à celle à vol d'oiseaux).
- Ce qu'il reste à faire :
 - génération aléatoire de trajets (utiliser une base de donnée des villes en France, sinon des points aléatoires dans le fichier des bornes) | Hélène

- finir programme simulation | Aurélien
 - sortie de la simulation (ce qu'on envoie pour l'affichage) | Aurélien
 - implémentation du multithreading dans la fonction de simulation | Mathis
 - affichage simulation (fichier txt, chaque ligne correspond à un tick, 1er caractère tick, un caractère séparateur entre bornes + données) | Adrien
- Rapport du projet :
- On va commencer à rédiger le rapport sur leaf. Aperçu des parties du rapport :
- état de l'art
 - partie trajet
 - partie simulation
 - tests (complexité mémoire et temps)

TO DO list :

Description	Personne(s) concernée(s)	Date
affichage simulation	Adrien	prochaine réu
finir partie simulation+gérer sorties	Aurélien	prochaine réu
multithreading	Mathis	prochaine réu
génération aléatoire de trajets	Hélène	prochaine réu
Rédaction du rapport	Tout le monde	24/05

Prochaine réunion : mercredi après-midi