

# Rendu projet intermédiaire

---

Anyak Aknine, Lucas Rouquairol

22/11/2022

## 1 Introduction

La première partie du projet consiste à mettre en place le réseau afin d'exécuter l'algorithme de coloration dans la deuxième partie. Chaque nœud est représenté par un processus qui travaille en parallèle des autres processus. Un parent jouera le rôle de serveur et créera / paramétrera les processus fils en fonction des instances passées en paramètre. Avant de nous lancer dans la structure actuelle, nous avons imaginé plusieurs autres solutions. Par exemple, une idée que nous avons abandonnée tard dans le développement est celle des threads pour la création des nœuds. Nous voulions créer un thread par nœud mais nous avons remarqué que les threads ne créent pas de nouveaux processus et c'est pour cela que nous avons abandonné cette solution.

## 2 Traduction des instances

Afin de sauvegarder les informations des instances, on commence par parcourir le fichier avec `fgets()` puis on stocke tous les voisins dans une matrice et le nombre de nœud. Une communication souhaitée entre deux processus est représentée par un 1 dans la matrice.

## 3 Création des processus

Pour créer les processus, on utilise la fonction `fork()` avec un processus parent qui aura le rôle de serveur et une boucle qui permet au parent de créer autant de processus fils que de nœud. On ajoute un attribut "idNode" qui sera différent pour chaque processus.

## 4 Suite du main

La suite du programme est différente en fonction du processus actuel. Le processus serveur commence par créer une socket sur laquelle il recevra les id des noeud et l'adresse de leur socket que chaque processus enverra afin de les stocker dans un tableau. Pour chaque ligne de notre matrice, le serveur envoie le nombre de voisins avec leur socket pour que chaque processus puisse mettre à jour ses attributs.

## 5 Ambiguïté du fork

Dans notre cas, on va donner les voisins a chaque processus avec un envoi de message par le serveur mais en vérité, chacun des processus connaît déjà la matrice des voisins car la traduction des informations est effectué avant la création des processus.

## 6 A terminer

Actuellement les processus ne connaissent pas encore les adresses de leur voisins mais il suffira de les envoyé avec le serveur sur chaque adresse reçu préalablement.