

Rapport de Tests – Projet C/C++

*Projet C/C++*

Sujet : « Réseaux sociaux : Recherche de composantes fortement connexes »

DREYER Quentin

JAMBET Pierre

NGUYEN Michael

Introduction :

Dans le cadre du projet de Langage C, nous avons effectué de nombreux « tests ». Nous détaillerons dans ce document, la nature de ces tests ainsi que leur intérêt pour le projet.

Voici la liste des tests que nous avons réalisés :

* Tests du programme sur une grande batterie de fichiers générés
  + Ces tests avaient pour but d’analyser les temps d’execution de notre programme en fonction du fichier analysé.
* Analyse des fuites mémoires et profilage avec VALGRIND
* Tests Unitaires réalisés avec le framework Google TEST

# Statistiques sur de nombreux tests

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre de sommet | Nombre de relations | Nombre de questions | Temps d’ouverture du fichier | Temps de recherche de CFC | Temps de recherche des distances |
| 2000 | 3 997 632 | 3 997 632 | 29, 515s | 4,103s | 1118, 79s |

A l’aide d’un simple script shell (voir ci-dessous) nous avons récolté des informations qui nous ont permis de réaliser des statistiques sur les performances du programme.

|  |
| --- |
| #!/bin/bash |
| # Script pour tester le programme avec pleins de fichiers generes. |
| # Premiere boucle : i => nombre de sommets |
| # Deuxieme boucle : j => nombre de relations |
| # Troisieme boucle : k => nombre de questions |
| date >> log |
| for (( i = 100; i < 2000; i+=100 )); do |
| echo -n $i " "; |
| for (( j = 1; j < 10; j+=1 )); do |
| for (( k=10; k < 1000; k+=20 )); do |
| ./GrapheFB --log $i $j $k >> log |
| done |
| done |
| done |
| echo ""; |

# Utilisation de VALGRIND

L’utilisation de valgrind nous a permis de tester la fiabilité et la robustesse de notre code. En utilisant le module memcheck nous avons pu vérifier le bon fonctionnement de notre programme au niveau des allocations mémoire et donc de la restitution de la mémoire. Pour effectuer de nombreux tests nous avons utiliser un script se basant sur la figure 1, mais dont la ligne principale est modifiée.

valgrind ./GrapheFB --log $i $j $k >> log2 2>&1

Ceci permet de rediriger les messages de valgrind dans le fichier log2, nous avons ainsi pu voir en parcourant ce fichier qu’à chaque exécution du programme, chaque zone mémoire allouée est libérée.

# Tests Unitaires avec Google Test