

# **Documentation de validation**

## **Projet Génie Logiciel 2022**

Salah-Eddine El Mehdi

Abidi Walid

Bakir Walid

El Khaldi Ahannach Yassin

Abdelouahab Mestassi

Janvier 14, 2022

### **1) Gestion de rendus et gestion de risques :**

Ce projet GL nécessite un grand volume de travail, donc une bonne gestion des rendus et des risques s'impose pour permettre aux membres du groupe de bien s'organiser, de se répartir les missions et d'atteindre l'objectif final.

#### **1.1) Les différentes dates de rendus :**

Au cours de ce projet nous serons amenés à rendre plusieurs rendus qui ont des dates limites différentes ,et donc y a un risque d'oublier une de ces dates .Pour cela l'équipe à prévu des dates de travail accéléré en équipe pour les taches critique (tel que le rendu intermédiaire ,le compilateur et les documentations du rendu) pour que ça soit prêt avant sa date limite .Ces dates importantes ainsi que les rendus ont bien été marqués sur le planning prévisionnel afin de les garder en vue et d'avoir une vue globale plus claire sur le projet .

#### **1.2) Compétence communication écrit :**

L'équipe s'assure toujours d'avoir une bonne communication entre les membres, afin d'échanger plus d'information et avancer plus rapidement cependant le choix de répartition des taches n'était pas parfaitement judicieux, et il fallait de temps en temps que plusieurs membres contribuent dans une seule partie .L'équipe n'a pas opter pour la préparation des documentations dès le début au fur et à mesure du travail sur le compilateur.

#### **1.3) Non respect des spécifications :**

Il est primordial de s'assurer que le compilateur (logiciel) soit conforme aux exigences du client, tel que les spécifications de la ligne de commande de **decac** doivent être respecté. Pour cela l'équipe fait des relectures à chaque fois du cahier des charges qui est notre poly du projet GL et pour chaque partie afin de s'assurer que chaque détail est respecté.

#### **1.4) Répartition des tâches non équilibré :**

Puisque on a opté pour la méthode d'agile, le volume de travail pendant chaque étape critique peut varier fortement selon chaque partie (étape A, B et C) ce qui peut engendrer une prise de retard sur le planning prévisionnel posé dès le début .Ainsi on fait des réunion régulières pour faire le point l'avancement et ce qui reste à faire pour ensuite répartir les taches de manière à atteindre le but avant sa date limite .

#### **1.5) Les tests non fiables :**

Les tests de la partie logicielle rédigée par le développeur risque de ne pas relever une erreur (possiblement grave), dans ce cas on aura de test biaisés .Pour cela, on essaye de faire relire les tests par d'autres membres, ou les rédiger par un autre que celui qui l'a implémenté.

#### **1.6) L'implantation de l'extension TRIGO :**

Il existe plusieurs approches pour implémenter les fonctions trigo, comme le développement illimité (DL) ou l'algorithme CORDIC qui s'avère une meilleur méthode pour calculer l'approximation qu'aux séries de Taylor .Cependant on est contraint par l'analyse énergétique qui est une partie importante du projet ,donc le but optimal est de trouver un équilibre entre la précision de calcul et le cout énergétique du produit