

Nom: LARGEOT
Prénom: Morgan
Promotion: 2027

Rapport d'expérience

Projet Raytracer

Projet effectué du 15 avril 2024 au 12 mai 2024



Introduction

Le projet Raytracer s'est déroulé dans le cadre du module de Programmation Orientée Objet, ayant pour objectif d'enseigner aux différents étudiants de l'école le paradigme de la programmation orientée objet, ainsi que des notions comme l'architecture et la gestion de projet.

Au travers de ce bilan, je vais décrire plus en détail mon parcours durant ce projet. Dans un premier temps, je vais faire une description générale du projet et dresser le bilan technique et organisationnel, ainsi que le bilan de qualité et des méthodes utilisées. Enfin, je donnerai mon appréciation générale concernant le projet.

Sommaire

1.DESCRPTION GÉNÉRALE DU PROJET

2.BILAN TECHNIQUE

3.BILAN DE L'ORGANISATION

4.BILAN QUALITÉ-MÉTHODES

5.APPRÉCIATION GLOBALE

Description générale du projet

Description Générale du Projet

Le projet consistait à développer un programme de ray tracing, une technique permettant de générer des images numériques réalistes en simulant le trajet inverse de la lumière. Le programme devait lire un fichier de configuration de scène et générer une image basée sur les descriptions fournies dans ce fichier.

Langage et Compilation

- Langage utilisé: C++
- Compilation: Via Makefile (avec les règles `all`, `clean`, `fclean`, `re`) ou CMake.

Fonctionnalités Obligatoires

Le programme devait inclure des primitives telles que les sphères et les plans, ainsi que des transformations comme la translation. Pour l'éclairage, il devait supporter la lumière directionnelle et la lumière ambiante, et les matériaux devaient être représentés par des couleurs uniformes. Le programme devait permettre de configurer la scène en ajoutant des primitives, en réglant l'éclairage et en configurant la caméra, avec une sortie d'image sous forme de fichier PPM sans interface graphique.

Format du Fichier de Scène

La scène à rendre devait être configurée dans un fichier externe, suggéré d'être au format libconfig++. Les étudiants avaient la liberté de structurer ce fichier comme ils le souhaitaient, en tenant compte de la configuration de la caméra, des primitives (sphères, plans), et des lumières (ambiante, ponctuelle, directionnelle).

Architecture

Les étudiants devaient utiliser des interfaces pour les primitives et les lumières pour permettre l'extensibilité du programme. Un système de plugins était également recommandé pour ajouter de nouvelles fonctionnalités sans réécrire le code de base. De plus, au moins deux patrons de conception (comme Factory, Builder, Composite, Decorator, Observer, State, Mediator) devaient être intégrés au projet.

Défense du Projet

Les étudiants devaient démontrer les fonctionnalités de leur raytracer lors d'une défense, en préparant des scènes et des captures d'écran de leur travail.

Organisation et durée du projet

Le projet devait être réalisé par groupe de 3 à 4 étudiants et avait une durée totale de 4 semaines, du 15 avril 2024 au 12 mai 2024.

Bilan Technique

Objectifs et résultat

Concernant l'atteinte des objectifs cités dans la description du projet, tous les objectifs ont été remplis, avec l'ajout de bonus tels que des primitives comme le tétraèdre, des couleurs UV, des transformations de mise à l'échelle, ainsi que l'import de scènes dans les fichiers de configuration. La présence obligatoire d'éléments liés à l'architecture du projet tels que le design pattern ou la présence de factory a également été prise en compte dans la réalisation du projet.

Problèmes rencontrés et solutions

Le projet s'est déroulé avec peu de problèmes dans l'ensemble, cependant quelques-uns étaient tout de même présents. En effet, des problèmes liés à l'interruption du processus de génération ou à des calculs mathématiques se sont manifestés durant le projet.

Pour y remédier, des solutions ont été apportées en intensifiant les recherches, en s'aidant mutuellement et en corrigeant les différents soucis que la création du code pouvait engendrer.

Points d'amélioration

Concernant les pistes d'amélioration possibles, cela pourrait passer par l'ajout de fonctionnalités. Au niveau des primitives, le tore pourrait être ajouté au projet final, de même que la transparence des matériaux, ou encore l'actualisation de l'image lors de modifications des fichiers de configuration.

Bilan Organisationnel

Communication et prise de décisions

Discord a été utilisé comme principal canal de communication, permettant des échanges fluides et la prise de décisions à travers des votes, favorisant ainsi une coordination efficace entre les membres du groupe.

La communication s'est également structurée autour d'une norme de commit commune et précise (norme karma).

Gestion des Tâches

GitHub Project a été utilisé pour la répartition des tâches, permettant d'assigner, de suivre et de prioriser les différentes activités du projet, assurant ainsi une répartition équilibrée du travail.

Documentation

Une documentation interne a été centralisée sur le site <https://raytracer.jeffroy.eu/>, fournissant aux membres du groupe mais aussi aux personnes extérieures, une référence essentielle pour les spécifications, les bonnes pratiques et les informations techniques du projet, favorisant ainsi l'autonomie et l'efficacité dans le travail.

Points d'amélioration

Afin de garantir le professionnalisme le plus absolu, un des points à améliorer serait d'opter pour une plateforme de communication plus professionnelle que discord.

Bilan Qualité-Méthodes

Qualité du Code

La qualité du code a été assurée par l'adoption de bonnes pratiques de programmation orientée objet. Chaque membre du groupe a suivi des normes strictes de codage, incluant la documentation des fonctions, le respect des conventions de nommage, et la mise en place de tests unitaires pour valider le bon fonctionnement des différentes fonctionnalités.

Revue de Code

Des revues de code régulières ont été effectuées pour identifier et corriger les erreurs, améliorer la qualité du code, et garantir que toutes les contributions respectent les standards de qualité du projet. Ces revues ont également favorisé le partage des connaissances entre les membres du groupe et l'amélioration continue du projet.

Appréciation Globale

Le projet Raytracer a été une expérience enrichissante, tant sur le plan technique qu'organisationnel. La mise en pratique des concepts de programmation orientée objet et des méthodes de gestion de projet nous a permis de développer un programme de ray tracing fonctionnel et extensible. Le travail en équipe a été facilité par l'utilisation d'outils de collaboration efficaces comme Discord et GitHub Project, et la documentation centralisée a joué un rôle clé dans le bon déroulement du projet.

En conclusion, ce projet nous a permis de renforcer nos compétences en programmation orientée objet, en gestion de projet, et en travail collaboratif, tout en réalisant un produit fini de qualité. Les défis rencontrés ont été surmontés grâce à une bonne organisation et à une coopération efficace, et les résultats obtenus ont dépassé nos attentes initiales.