

Projet – Langages à Objet Avancés

Auteurs :

- XUE Guillaume 22101031
- YU David 22110478

Introduction :

Ce projet consiste à recréer le jeu « Piece Out » en utilisant la bibliothèque SFML pour la gestion graphique en c++. Le jeu implique des pièces qui peuvent être déplacées, tournées et inversée (symétrie) sur une carte. Le projet est structuré en plusieurs modules, chacun responsable d'une partie spécifique de la logique du jeu.

Aspects Traités

Structure du Projet

Src/

 Controllers/

 Models/

 Utils/

 Views/

Contrôleurs

- GameController : Gère le cycle de vie du jeu.
- HeaderController : Gère les interactions avec l'en-tête de l'interface utilisateur.
- MenuController : Gère les interactions avec le menu.
- MouseController : Gère les interactions de la souris.
- KeyboardController : Gère les interactions du clavier.
- PieceController : Gère les interactions avec les pièces.

Modèles

- Maps : Représente la carte du jeu.
- Piece : Représente une pièce du jeu.
- PieceOperateur : Représente les opérations possibles sur une pièce (déplacement, rotation, symétrie).

Vues

- MapVue : Affiche la carte du jeu.
- PieceVue : Affiche une pièce du jeu.
- HeaderVue : Affiche l'en-tête de l'interface utilisateur.
- MenuVue : Affiche le menu du jeu.

Utilitaires

- **Command** : Interface pour les commandes exécutables.
- **CommandManager** : Gère l'exécution et l'annulation des commandes.
- **Observer** : Interface pour le pattern Observer.
- **Subject** : Interface pour le pattern Subject.

Aspects Significatifs

Utilisation de SFML

Le projet utilise la bibliothèque SFML pour la gestion graphique, ce qui permet de créer une interface utilisateur interactive et « visuellement attrayante ».

Pattern de Conception

Le projet utilise plusieurs patterns de conception, notamment :

- **Singleton** : Utilisé pour les contrôleurs afin de garantir qu'il n'y a qu'une seule instance de chaque contrôleur.
- **Observer** : Utilisé pour mettre à jour les vues en réponse aux interactions de l'utilisateur.
- **Command** : Utilisé pour gérer les actions des pièces et permettre leur annulation.
- **Factory** : Utilisé pour créer des instances de pièces et d'opérations de manière flexible.
- **Visitor** : Utilisé pour ajouter de nouvelles opérations aux pièces sans modifier leurs classes et permettre un accès plus « naturel » des classes.
- **Decorator** : Utilisé pour ajouter dynamiquement des responsabilités supplémentaires aux objets de pièces.

Gestion des pièces

Les pièces peuvent être déplacées, tournées et inversées(symétrie). Chaque opération est représentée par une classe spécifique (OperateurDeplacement, OperateurRotation, OperateurSymetrie), ce qui permet une gestion flexible et extensible des opérations sur les pièces.

Problèmes Connus

Gestion de la mémoire

Il y a des risques de fuites de mémoire, notamment lors de la création et de la suppression des pièces et des commandes, la partie liée à SFML n'a pas pu être corrigée.

Vérification des Mouvements & gestion fin de partie

La vérification des mouvements des pièces peut être améliorée pour éviter les mouvements invalides. Actuellement, la vérification est basique et lourde, certes ça prend en compte tous les cas possibles mais c'est une vérification qui doit copier puis vérifier tout la map avec toutes les pièces une par une.

Interface Utilisateur

L'interface utilisateur est fonctionnelle mais pourrait être améliorée pour être plus intuitive et attrayante, actuellement il va à l'essentiel.

Extensions Non Implémentées

Niveaux supplémentaires

Le projet pourrait être étendu avec des niveaux supplémentaires pour offrir plus de défis aux joueurs. Actuellement, seuls quelques niveaux sont implémentés (1, 2, 3 respectivement les parties présentées dans le sujet, 8 et 9 sont des tests).

Résolution automatique

On pourrait intégrer une méthode qui pourrait calculer à chaque instant de la carte les meilleurs déplacements possibles pour terminer la carte.

Carte aléatoire

De même que pour la résolution automatique, là ce serait le cas pour la création de la carte sans qu'on ait besoin d'intervenir, l'équivalent de la carte « défi quotidienne » du jeu.

Classement Multijoueur

C'est une extension qui irait avec le « défi quotidien » où on enregistre le meilleur temps et le meilleur nombre de coup pour ce défi et on propose un classement avec tout les participant.