

Colors Crush

Table des matières :

1. Introduction

- 1.1. Présentation du projet
- 1.2. Objectifs pédagogiques

2. Description des fonctionnalités

- 2.1. Page d'accueil principale du jeu
- 2.2. Page de sélection du mode de jeux
- 2.3. Écran affichant les règles du jeu
- 2.4. Exemple de chargement depuis un fichier
- 2.5. Capture du jeu en cours
- 2.6. Écran de fin de partie

3. Structure du programme

- 3.1. Classes Principales
- 3.2. Composants Graphiques
- 3.3. Gestion des Évènements
- 3.4. Boutons et Navigation

4. Données représentant l'état de la partie

- 4.1. Grille de Jeu
- 4.2. Score Courant
- 4.3. État de la Partie

5. Conclusion personnelle

- 5.1. Conclusion d'Amine
- 5.2. Conclusion de Lucas

Introduction :

Dans le cadre de ce projet, nous avons développé un jeu intitulé Colors Crush, un jeu de réflexion dans lequel le joueur doit vider une grille remplie de blocs colorés. Colors Crush met en avant des mécaniques simples mais addictives : le joueur sélectionne et élimine des groupes de blocs adjacents de même couleur, ce qui entraîne un réarrangement automatique des blocs restants afin de combler les espaces vides. L'objectif est d'obtenir le score le plus élevé possible en supprimant de grands groupes de blocs simultanément.

Le projet est réalisé en langage Java, sans dépendances extérieures, en s'appuyant uniquement sur les composants graphiques et les bibliothèques standards fournies par l'API officielle Java. L'interface du jeu est entièrement interactive, contrôlée exclusivement à l'aide de la souris.

Description des fonctionnalités :

1. Page d'accueil principale du jeu :



Description :

Écran d'accueil principal du jeu. Trois boutons :

- **Jeux** (rouge)
- **Règles** (bleu)
- **Quitter** (vert)

2. Page de sélection du mode de jeu :

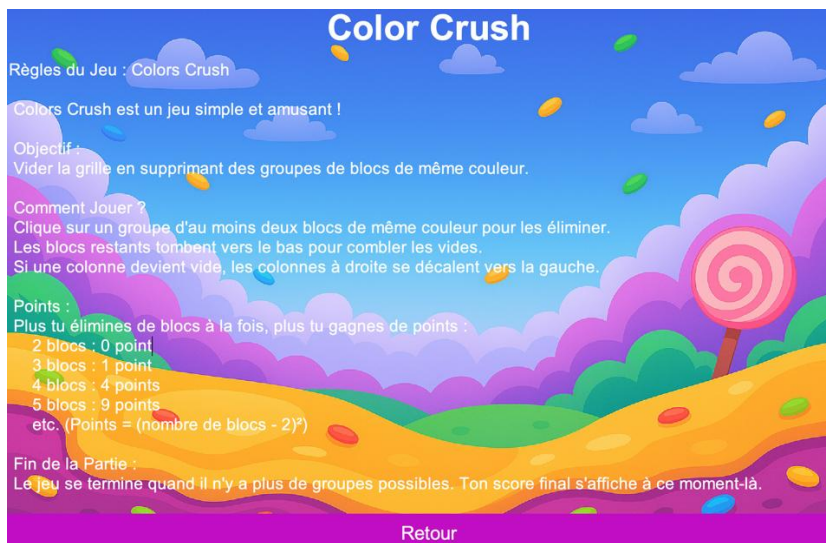


Description :

Écran permettant de choisir comment commencer une nouvelle partie :

- **Jouer** (grille aléatoire)
- **Charger** (chargement d'une grille spécifique via un fichier)
- **Retour** (retour à l'écran précédent)

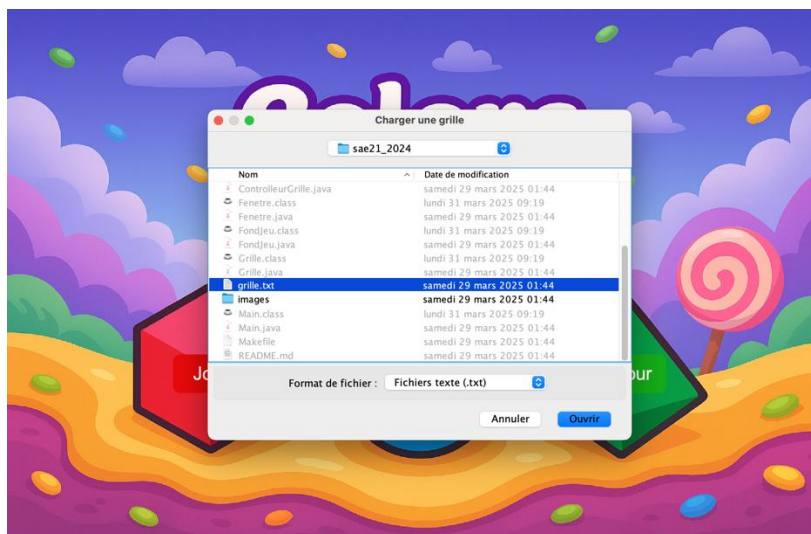
3. Écran affichant les règles du jeu :



Description :

Écran clair et détaillé expliquant brièvement comment jouer à Colors Crush.
Contient les règles et le calcul des points.

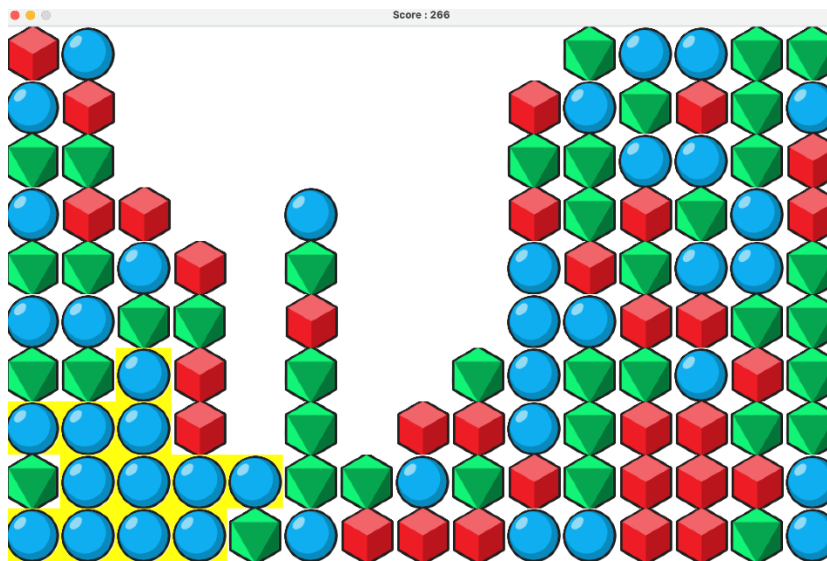
4. Exemple de chargement depuis un fichier :



Description :

Exemple de la fenêtre JFileChooser permettant de sélectionner un fichier pour charger une grille prédéfinie.

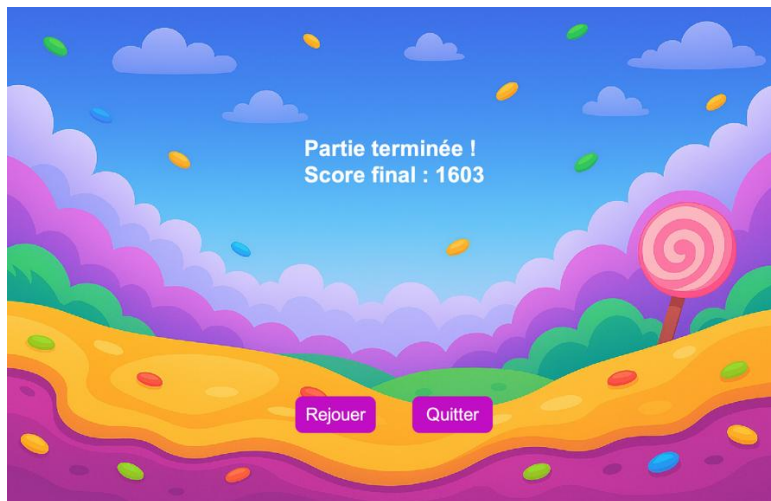
5. Capture du jeu en cours :



Description :

Exemple concret d'une partie en cours, avec la grille de blocs colorés. On voit l'accentuation d'un groupe lorsqu'il est survolé par la souris. Le score actuel est affiché en haut.

6. Écran de fin de partie :



Description :

Message clair indiquant la fin du jeu, affichant le score final et proposant les options :

- **Rejouer** (retourne a la page d'accueil principale du jeu)
- **Quitter**

Structure du Programme

Le programme Colors Crush est organisé de façon modulaire afin de garantir lisibilité et maintenabilité. Chaque classe a un rôle précis et distinct, contribuant au bon fonctionnement global du jeu.

1. Classes Principales :

- **Main.java :**
 - Point d'entrée du programme.
 - Initialise l'interface principale du jeu.
- **Acceuil1.java et Acceuil2.java :**
 - Gestion des écrans d'accueil permettant le choix du type de partie (aléatoire ou chargée depuis un fichier).
- **Regles.java :**
 - Affichage clair des règles du jeu à l'utilisateur.
- **Jeu.java, JeuAleatoire.java et JeuFile.java :**
 - Classes qui définissent la logique principale du jeu, incluant la gestion des différents modes de démarrage de partie.

2. Composants Graphiques :

- **Grille.java :**
 - Génère la grille de jeu, constituée de blocs de couleur.
 - Assure le rafraîchissement visuel après chaque action du joueur.
- **Bloc.java :**
 - Représentation individuelle d'un bloc sur la grille.
 - Gère l'apparence et l'état (couleur, vide ou non).
- **FondJeu.java :**
 - Fournit un arrière-plan personnalisé et esthétique pour l'interface du jeu.

3. Gestion des Événements :

- **ControlleurGrille.java :**
 - Gère l'interaction utilisateur avec la grille : survol, sélection, suppression et réorganisation des blocs.
 - Contient l'algorithme de détection de groupes de blocs.
- **FinJeu.java :**
 - Gère la détection de la fin de partie.
 - Affiche le score final et propose au joueur de rejouer ou de quitter.

Données Représentant l'État de la Partie

L'état actuel d'une partie de Colors Crush est représenté principalement par les données suivantes :

1. Grille de Jeu

La grille de jeu constitue le cœur du jeu, structurée comme suit :

- **Matrice 2D (tableau de blocs) :**
 - Taille fixe de 10 lignes par 15 colonnes.
 - Chaque cellule contient un objet de type Bloc.

Chaque bloc est caractérisé par :

- **Couleur** : Rouge ('R'), Vert ('V') ou Bleu ('B').
- **Position** : Coordonnées (ligne, colonne) dans la grille.
- **État** : Vide ou rempli.

2. Score Courant

- **Valeur numérique entière** indiquant les points accumulés au fil de la partie.
- Calculé selon la règle $(n - 2)^2$, avec n représentant le nombre de blocs éliminés simultanément.

3. État de la Partie

- **Booléen indiquant si la partie est terminée ou non :**
 - Partie en cours : le joueur peut encore effectuer des actions.
 - Partie terminée : aucun groupe de blocs adjacents disponible, affichage du score final.

Conclusion personnelle :

Amine :

Ce projet m'a permis d'appliquer un algorithme de parcours en largeur pour gérer les groupes de blocs, tout en renforçant mes bases en mathématiques et en structures de données. J'ai aussi appris à structurer un projet Java efficacement et à manipuler les composants graphiques pour créer une interface interactive et fluide.

Lucas :

Ce projet m'a permis de renforcer mes compétences en programmation Java et d'approfondir mes connaissances en conception modulaire et gestion d'interface graphique. J'ai apprécié particulièrement la mise en œuvre de l'algorithme de détection de groupes, qui représentait un défi stimulant.