

SUPINFO – Projet de fin d'année

Documentation Technique

Projet : TWELVE MEN'S MORRIS

Étudiant : Aguibou Sow

École : SUPINFO

Année académique : 2024 – 2025

Introduction

Le jeu fonctionne sur les systèmes Windows, Mac Ce document présente la documentation technique du projet *Twelve Men's Morris*, un jeu de plateau développé dans le cadre du projet de fin d'année à SUPINFO.

Le but du projet est de proposer une version bureau complète de ce jeu stratégique traditionnel, en respectant les règles officielles. Le développement a été réalisé en Python, avec l'utilisation de la bibliothèque Tkinter pour la création de l'interface graphique.

OS et Linux. Il propose une interface interactive, un tour par tour, la gestion des phases du jeu et la détection des conditions de victoire.

Objectifs du projet

- Comprendre les règles du jeu Twelve Men's Morris et les modéliser en code.
- Implémenter une interface graphique fonctionnelle avec Tkinter.
- Décomposer le jeu en plusieurs phases de logique (pose, mouvement, saut).
- Offrir une expérience de jeu fluide et fidèle aux règles.
- Documenter toutes les étapes du développement pour permettre la lecture, la maintenance et l'évolution du projet.

Technologies utilisées

- **Langage** : Python 3
- **Bibliothèque graphique** : Tkinter
- **Éditeur de code** : Visual Studio Code
- **Plateformes visées** : Windows, Mac OS, Linux

Fonctionnalités développées

1. Création du plateau

Le plateau a été modélisé sous forme de grille de coordonnées avec des emplacements valides. Chaque case correspond à une position définie, ce qui permet une gestion précise des actions du joueur.

2. Phase 1 – Pose des pions

Chaque joueur dispose de 12 pions. À tour de rôle, ils posent un pion sur une case vide du plateau.

La logique mise en place :

- Vérification que la case est libre.
- Vérification du tour du joueur.
- Compte du nombre total de pions posés.

3. Détection des moulins

Un moulin est formé lorsqu'un joueur aligne trois de ses pions horizontalement ou verticalement. Le système détecte automatiquement cette situation après chaque pose ou déplacement.

4. Suppression d'un pion adverse

Lorsqu'un joueur forme un moulin, il peut supprimer un pion adverse. Cette suppression est conditionnée : il est interdit de supprimer un pion faisant partie d'un moulin actif, sauf si aucun autre pion n'est disponible.

5. Passage automatique à la phase suivante

Une fois les 24 pions posés (12 par joueur), le jeu passe automatiquement à la phase 2 : le mouvement des pions.

6. Détection de la victoire

Le jeu détecte une fin de partie lorsque :

- Un joueur possède moins de 3 pions.
- Un joueur ne peut plus effectuer de déplacement valide.

- Le système affiche alors un message de victoire ou d'égalité et propose de recommencer.

7. Gestion des erreurs et limitations

Les pions ne peuvent pas être posés sur une case occupée.

Aucun joueur ne peut dépasser les 12 pions.

Les clics sont bloqués si une suppression est en cours ou si le jeu est terminé.

Phase 2 – Mouvement des pions

Une fois la phase de pose terminée, les joueurs déplacent leurs pions à tour de rôle.

Fonctionnement :

- Le joueur clique sur l'un de ses pions déjà placés.
- Puis clique sur une case vide adjacente (voisine).
- Si le déplacement est valide, le pion est déplacé.
- Si un moulin est formé, le joueur peut à nouveau supprimer un pion adverse.

Une fonction dédiée vérifie si la case cible est une voisine directe de la position initiale.

Cette phase est stratégique car elle permet aux joueurs de construire des moulins ou de bloquer ceux de l'adversaire.

Phase 3 – Le saut (vol libre)

Lorsqu'un joueur tombe à trois pions, il peut les déplacer librement sur n'importe quelle case vide du plateau, sans contrainte de voisinage.

Cette règle spéciale a été ajoutée pour permettre au joueur en difficulté de continuer à jouer. Elle rend le jeu plus équilibré.

Le code détecte automatiquement cette situation et autorise le saut libre uniquement pour le joueur concerné.

Conditions de fin du jeu

Le jeu se termine automatiquement dans les cas suivants :

- Un joueur a moins de trois pions sur le plateau.
- Un joueur ne peut plus se déplacer.
- Aucun mouvement possible pour les deux joueurs (match nul).

Dans chaque cas, un message est affiché et un bouton permet de recommencer une nouvelle partie.

Choix techniques

- Tkinter a été choisi pour sa simplicité, sa portabilité et son intégration directe dans Python.
 - Les actions de jeu sont gérées dans une seule fonction centrale cliquer(event) qui interprète les clics selon l'état actuel du jeu.
 - Des variables globales contrôlent l'état du jeu : phase, joueur actif, pions restants, etc.
 - Le code est commenté de manière simple, sans complexité inutile, pour faciliter la compréhension et la relecture.
 - Chaque fonctionnalité a été validée progressivement afin de corriger les bugs au fur et à mesure et assurer la stabilité du jeu.
-

Conclusion

Ce projet m'a permis d'appliquer mes connaissances en programmation Python dans un cadre concret et complet. Le fait de devoir structurer un jeu par étapes m'a appris à raisonner comme un développeur, à anticiper les erreurs, à organiser mon code, et à construire une application graphique interactive.

Le travail est encore en cours, notamment sur la finition des phases avancées et sur l'amélioration de l'interface, mais les bases sont solides et fonctionnelles.

Ce projet représente pour moi une étape importante dans ma progression, car je l'ai réalisé entièrement en comprenant chaque ligne de code. La documentation me permet de garder une trace claire de mon raisonnement et de mon avancement.