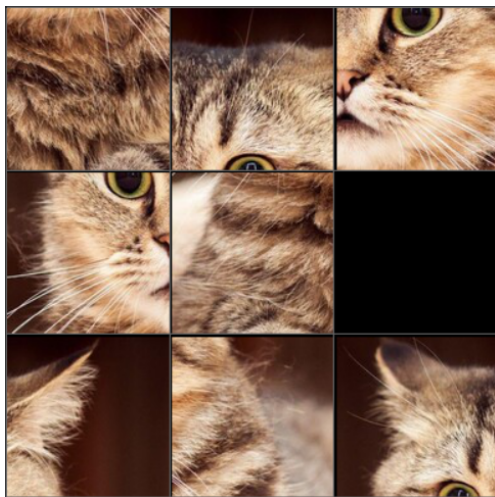


Université Galatasaray
Département de Génie Informatique
INF444 - Intelligence Artificielle

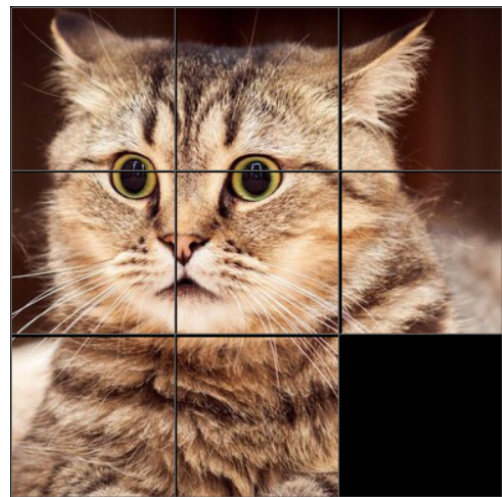
Projet

Au cours de ce projet, vous allez concevoir et modéliser un agent intelligent pour résoudre une instantiation simple du problème reconnu de N-Puzzle. Le jeu de N-Puzzle est composé de $n \times n - 1$ tuiles qui glissent dans un cadre du format $n \times n$ avec une case vide permettant de modifier la configuration des tuiles. Le jeu consiste à remettre dans l'ordre ces tuiles à partir d'une configuration initiale quelconque.

Comme projet, vous avez un problème de 8-puzzle composé d'une image projetée sur une grille de 3×3 comme illustré sur la Figure 1a. Le but est de déplacer ces tuiles et trouver la bonne combinaison pour assembler l'image, comme le montre la Figure 1b.



(a) Une configuration initiale



(b) Image recomposée

Figure 1: Le jeu de 8-Puzzle avec l'image d'un chat choqué

L'environnement de jeu vous est fourni avec les fichiers Python sur Moodle (`skeleton.zip`). L'interface du jeu est lancée par l'exécution du fichier de `main.py`. Vous pouvez jouer au jeu vous-même à l'aide de votre curseur après avoir mélangé les tuiles.

Le jeu a une interface utilisateur simple, comme illustré sur la Figure 2. Vous avez des boutons:

- pour changer l'image (1, 2, 3)
- pour mélanger les tuiles aléatoirement (Shuffle)
- pour mélanger les tuiles en faisant 5, 10, et 15 déplacements (Shuffle 5, 10, 15)
- pour résoudre le problème (Solve Puzzle)
- pour réinitialiser le jeu (Reset)

Vous avez aussi le temps qui s'écoule et le meilleur score obtenu, affichés sur l'écran. Le score est relié au temps d'exécution de votre algorithme, c-à-d le temps qui passe pour résoudre le problème et atteindre l'objectif.

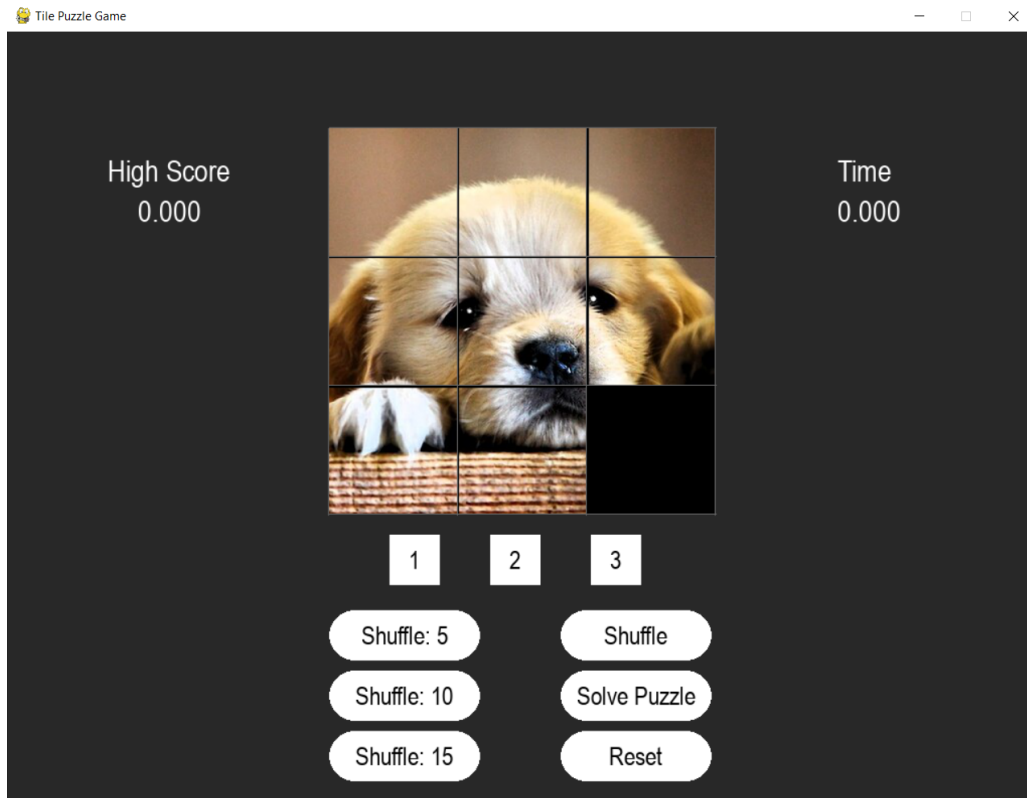


Figure 2: Interface utilisateur de notre jeu

Instructions :

- Ce n'est pas un travail individuel, c'est un travail d'équipe, vous devez donc travailler en équipe de 2 ou 3 étudiants.
- Vous allez écrire le code de votre agent dans le fichier `agents.py`, vous pouvez ajouter de nouvelles fonctions à ce fichier si nécessaire (mais faites attention le bouton "Solve Puzzle" fait appel à la fonction `solve_puzzle`).
- Vous pouvez utiliser n'importe quelle fonction et fichier de la bibliothèque Python standard et des bibliothèques fournies telles que numpy.
- Vous n'êtes pas autorisé à modifier les autres fichiers. Vous pouvez modifier les variables `GAME_SIZE` et `SHUFFLE` dans le `settings.py` pour expérimenter, mais soyez prudent, si on augmente la valeur de ces variables ça pourrait créer des jeux insolubles.
- Vous allez rendre seulement `agents.py` accompagné d'un rapport EN FRANÇAIS pendant la remise de vos projets. Dans votre rapport,
 - Décrivez l'algorithme que vous avez conçu pour résoudre ce problème.
 - Expliquez pourquoi vous avez choisi d'implémenter cet algorithme.
 - Listez quels sont les avantages et les inconvénients d'utiliser cet algorithme.
 - Rapportez vos résultats et comparez les performances de votre algorithme avec celles des autres algorithmes en général, en commentant sur la complétude, l'optimalité et les complexités temporelle et spatiale.
 - Notez le meilleur score que vous avez obtenu.
 - Même si vous n'arrivez pas à trouver une solution pour le problème, répondez aux questions précédentes et rédigez votre rapport. Expliquez pourquoi vous avez échoué à votre implémentation.
- La date limite de remise des projets est **17 Janvier 2024 (23h59)**.