

Implémentation de l'algorithme de Dijkstra et sa variante A^* en Python 3 et Tkinter

Boisnier Thomas

13 mai 2018

1 Algorithmes

- Composants
- Distance
- Heuristique
- Déroulement

2 Historique

- Principe

3 Tkinter

- Organisation

4 Progression

L'algorithme de Dijkstra et sa variante A* nécessite plusieurs éléments :

- un point de départ
- un point d'arrivé
- un calcul de distance : coût G
- une heuristique : coût H (uniquement pour A*)
- une liste dite fermée, qui contiendra les points visités
- une liste dite ouverte (file à priorité), qui contiendra les points accessibles
(tri croissant du coût total : coût $F = \text{coût } G + \text{coût } H$)

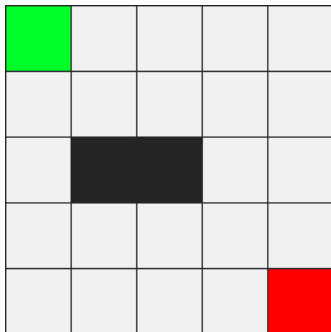
Calcul de la distance (coût G) entre 2 points :

- 1 dans toutes les directions
- 1 en horizontale et verticale, $\sqrt{2}$ en diagonale

Si A* utilise une heuristique (coût H) qui ne surestime jamais la distance du but, A* peut être avéré admissible.

```
1 dx = abs(node.m_x - target.m_x)
2 dy = abs(node.m_y - target.m_y)
```

Manhattan	$dx + dy$
Chebyshev	$(dx + dy) - \min(dx, dy)$
Octile	$(dx + dy) + (\sqrt{2} - 2) * \min(dx, dy)$
Euclidean	$\sqrt{dx * dx + dy * dy}$



● Case courante



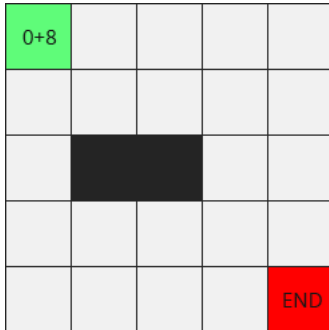
Liste ouverte



Liste fermée



Obstacle



● Case courante



Liste ouverte



Liste fermée



Obstacle



● Case courante



Liste ouverte



Liste fermée



Obstacle

0+8	1+7	2.8+6		
1+7	1.4+6	2.4+5		
2.8+6				
				END

● Case courante



Liste ouverte



Liste fermée



Obstacle

	1+7	2.8+6	3.8+5	
	1.4+6	2.4+5	3.4+4	
			3.8+3	
				END

● Case courante



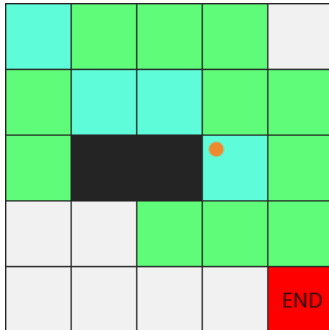
Liste ouverte



Liste fermée



Obstacle



● Case courante



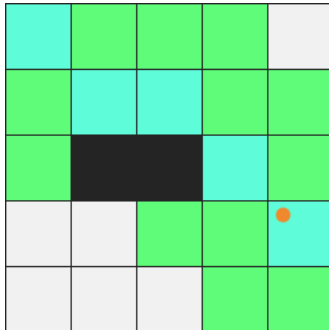
Liste ouverte



Liste fermée



Obstacle



● Case courante



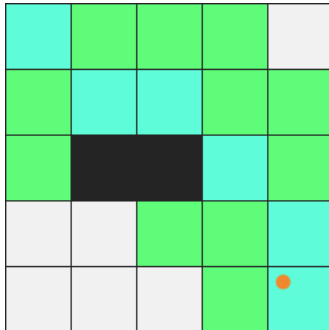
Liste ouverte



Liste fermée



Obstacle



● Case courante



Liste ouverte



Liste fermée



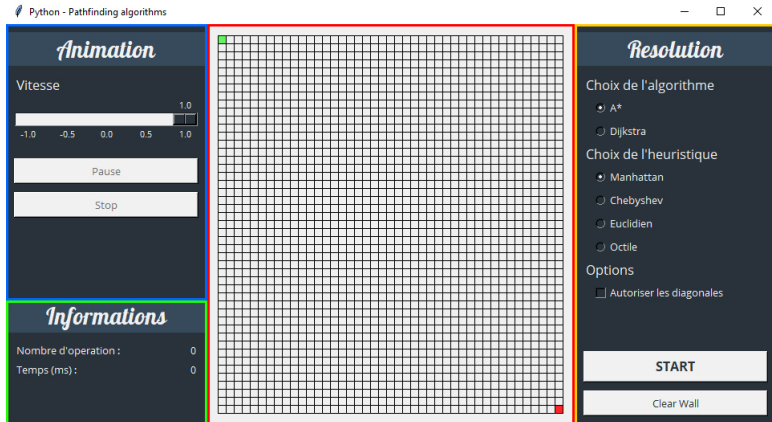
Obstacle

Historique des actions importantes.

- Historique : Liste contenant les actions
- Action : Structure contenant les informations que l'on souhaite (temps, type, ...)

Appel d'une fonction au cours du déroulement de l'algorithme qui ajoute une action à l'historique.

```
1 history.add(action_type, cell, time)
```



Composants : Menu d'animation (bleu), Cadre des informations (vert), Grille (rouge), Menu de choix de résolution (jaune)

Organisation de l'avancement du projet :

- Documentation
- Création de la grille et avec le menu de résolution
- Gestion des événements (déplacement des cases de départ/arrivée, création/suppression des obstacles)
- Implémentation de l'algorithme A* (rendu en temps réel)
- Implémentation d'un historique
- Gestion de l'animation
- Ajout d'un fichier de configuration