

Intelligence artificielle



Rime Haddad

Roman Rieunier

Groupe 2

Introduction

Ce rapport a pour but de présenter le projet d'Intelligence artificielle réalisé lors de notre deuxième année d'étude à l'ENSC.

La première partie de ce rapport est consacrée à ce que nous avons réalisé durant le projet et aux méthodes adoptées (élaboration des questions, de l'interface, ...).

La deuxième partie présente quant à elle la gestion du projet, le rôle de chacun, les outils utilisés durant le projet.

Enfin, nous avons mis en annexe le questionnaire réalisé, et transmis notre code source compressé au format zip.

RÉALISATIONS

a. Questions

Lors de ce projet nous avons traité deux parties.

En ce qui concerne la première partie, nous avons réalisé des winforms dans lesquels nous y avons mis des questions en relation avec le cours et les TP d'intelligence artificielle de cette année. Pour le choix des questions, nous avons fait en sorte de balayer tout le cours, et notamment les notions les plus importantes telles que l'algorithme de Dijkstra, du MinMax, d'A*. Les questions portent principalement sur la simple connaissance du cours. Toutefois, certaines questions portent sur la mise en pratique des méthodes acquises en cours et en TP, comme par exemple la résolution d'un problème via l'algorithme de Dijkstra, ou encore la recherche des degrés de liberté d'un robot.

Toutes les questions valent 1 point à l'exception de la question portant sur l'algorithme de Dijkstra évaluée sur 3 points. Il n'y a pas de points négatifs en cas de mauvaise réponse. Néanmoins, certaines questions ont plusieurs réponses correctes. Si une des réponses correctes n'est pas présente (ou si la personne répond une réponse correcte et une réponse fausse), aucun point n'est attribué.

Certaines méthodes vues en cours sont très proches, c'est pourquoi nous avons fait le choix de poser des questions permettant d'identifier si oui ou non, l'étudiant avait compris les différentes notions du cours, sans confusion.

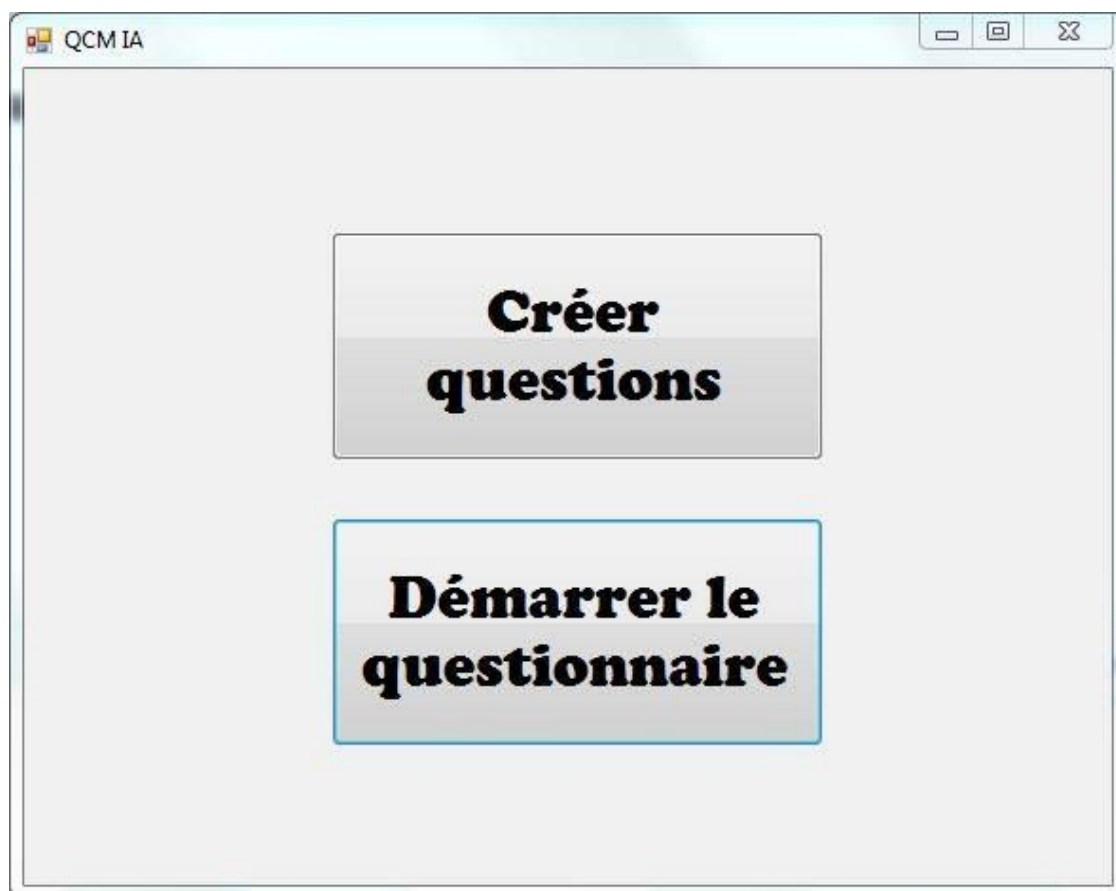
b.Interface

Après la rédaction des questions, nous nous sommes occupés de la partie interface de notre questionnaire. Pour ce faire, nous avons créé des Winforms sur Visual Studio.

L'idée principale était de rendre l'interface du questionnaire ergonomique, facile d'utilisation. C'est pourquoi nous avons par exemple choisi des boutons cliquables.

Voici des captures d'écran montrant le fonctionnement de notre questionnaire.

Tout d'abord, l'étudiant souhaitant répondre au questionnaire peut cliquer sur le bouton "Démarrer le questionnaire".



Ensuite, après avoir lancé le questionnaire, des questions en haut à gauche s'affiche sur l'écran. Pour répondre, il suffit de cocher les cases situées dans la listbox et d'appuyer sur le bouton valider. Comme on peut le voir ci dessous, une question peut avoir une ou plusieurs réponses.

Le score s'affiche en haut à droite.

Questionnaire

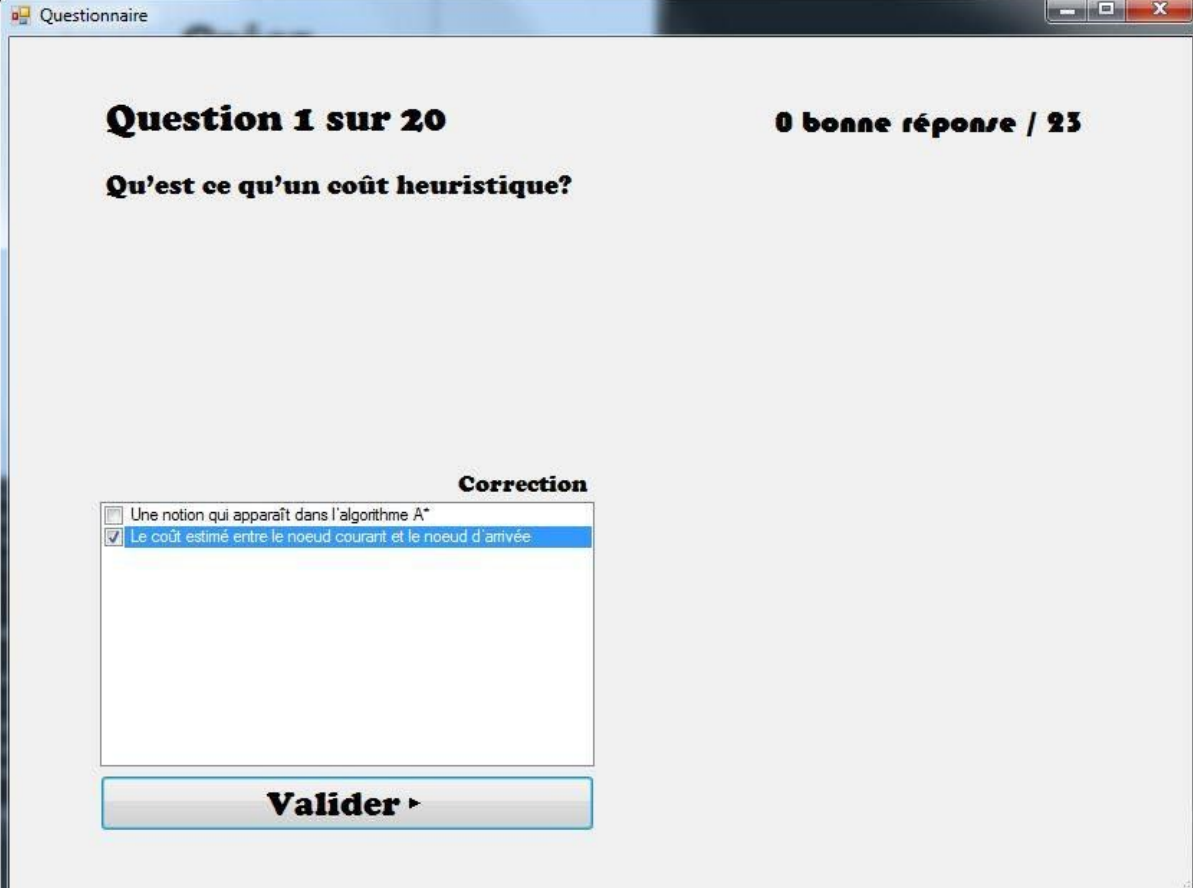
Question 5 sur 21 **4 bonnes réponses / 23**

En quoi le langage humain est-il problématique dans l'élaboration d'une intelligence artificielle ?

- ☒ La syntaxe
- ☐ La sémantique
- ☒ L'intonation
- ☐ La langue

Valider ►

Lorsque la personne s'est trompé de réponse, la réponse correcte apparaît par la suite dans la correction.



The screenshot shows a window titled "Questionnaire" with standard Windows window controls. The main content area displays "Question 1 sur 20" and "0 bonne réponse / 23". Below this is the question "Qu'est ce qu'un coût heuristique?". A "Correction" section follows, containing two radio button options. The first option, "Une notion qui apparaît dans l'algorithme A*", is unselected. The second option, "Le coût estimé entre le noeud courant et le noeud d'arrivée", is selected and highlighted with a blue background. Below the options is a large empty text input field. At the bottom of the window is a button labeled "Valider ►".

Question 1 sur 20 **0 bonne réponse / 23**

Qu'est ce qu'un coût heuristique?

Correction

☐ Une notion qui apparaît dans l'algorithme A*

☒ Le coût estimé entre le noeud courant et le noeud d'arrivée

Valider ►

Enfin, voici à quoi ressemble notre exercice sur l'algorithme de Dijkstra. Un graphe est sélectionné au hasard parmi les deux que l'on a codés. Ici nous avons opté pour des boutons cliquables, qui s'adaptent au nombre de nœuds, permettant la saisie du chemin à parcourir. Les fermés et les ouverts sont représentés ainsi que l'arbre final, à droite. Cette question vaut trois points. Un point est donné pour la résolution des ouverts, un pour les fermés, et le dernier pour la représentation de l'arbre final.

Algorithmme Dijkstra

Appliquez l'algorithme de Dijkstra à ce graphe

Fermés

{ }

{ A }

{ }

A	B
C	D
E	F
G	H

Valider ligne

Ouverts

{ A }

{ B, D, C }

{ }

B	A
D	C
F	E
H	G

Valider

Arbre final

Valider

Valider >

RÉPARTITION DES TÂCHES

Durant ce projet, nous avons travaillé en deux temps. D'une part nous avons débuté le projet par la rédaction des questions. Chacun s'est occupé de dix questions, balayant l'ensemble des cours en intelligence artificielle, mais aussi des TP effectués. Les questions ont d'abord été rédigées sur un document Google Drive.

Ensuite, une fois les questions rédigées, nous avons travaillé sur la réalisation des Winforms.

La plupart du temps nous avons adopté la méthode du pair programming. Néanmoins, lorsque nous travaillions sur deux postes différents, nous passions nécessairement par Github.

Une fois la première partie achevée, nous nous sommes concentrés sur l'élaboration de l'algorithme de Dijkstra. Tandis que Roman effectuait les tests finaux et optimisait l'algorithme de Dijkstra, Rime rédigeait le rapport du projet.

Pour nous organiser durant ce projet, nous avons utilisé Trello pour nous attribuer les tâches principales à effectuer. De manière générale, ce projet nous a permis de mieux comprendre à la fois le cours, mais aussi la création des Winforms.

Annexes

Annexe 1 : QUESTIONNAIRE

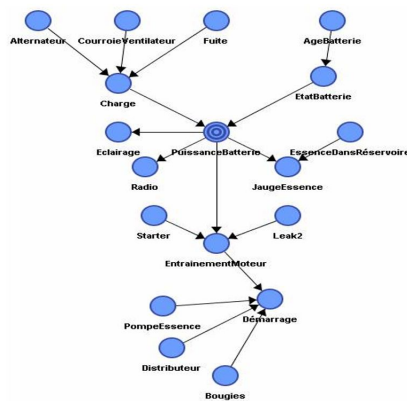
Cette partie présente les différentes questions posées dans l'exercice 1. Chaque réponse correcte est affichée en *gras italique*. Certaines questions ont plusieurs réponses correctes. Les questions sont notées sur un point, la question sur Dijkstra vaut quant à elle 3 points.

1. Qu'entend t-on par IA faible ?
 - a. La conception des programmes qui auront autant de conscience et de personnalité que des humains.
 - b. La conception de programmes partiellement intelligents, où l'on ne peut pas obtenir la conscience et la pensée de façon algorithmique.***
 - c. Une interface non automatisée
 - d. Ce terme n'existe pas.
2. Combien ce robot a t-il de degrés de liberté ?
 - a. 6***
 - b. 7
 - c. 10
 - d. 13



3. Laquelle de ces propositions ne constitue pas un sous-domaine de l'IA ?
 - a. La reconnaissance visuelle
 - b. Les algorithmes génétiques
 - c. *L'étude des facultés mentales humaines*
 - d. Les systèmes experts
4. En quel année est apparu le terme robot pour la première fois ?
 - a. 1919
 - b. *1920*
 - c. 1921
 - d. 1922
5. En robotique, que permet une liaison pivot ?
 - a. La rétraction
 - b. La propulsion
 - c. La translation
 - d. *La rotation*
6. Quel est l'avantage d'un microcontrôleur ?
 - a. *L'autonomie*
 - b. La rapidité
 - c. L'adaptabilité
 - d. L'utilisabilité
7. Le capteur LIDAR est utilisé pour :
 - a. Les radars
 - b. Prévenir les risques sismiques
 - c. *Détecter les obstacles*
 - d. Capter les ondes sonores
8. Quelle est l'hypothèse sous-jacente à l'algorithme alpha beta ?
 - a. L'adversaire est meilleur que soi
 - b. L'adversaire joue au hasard
 - c. L'adversaire a plus de chance de perdre
 - d. *L'adversaire joue au mieux*
9. En quoi le langage humain est-il problématique dans l'élaboration d'une intelligence artificielle ?
 - a. *La syntaxe*
 - b. *La sémantique*
 - c. L'intonation
 - d. La langue

10. En linguistique, qu'est ce qui n'est pas vrai ?
- a. Chien est un hyponyme d'animal
 - b. Dijkstra est un hyponyme d'algorithme
 - c. Empereur est un hyperonyme de Napoléon
 - d. *Curiosity est un hyperonyme de robot*
11. A quoi servent les algorithmes génétiques ?
- a. *A résoudre des problèmes d'optimisation*
 - b. A mieux comprendre les phénomènes génétiques
 - c. A analyser les variables physiologiques
 - d. A étudier les facultés mentales humaines
12. Que représente cette image ?
- a. Diagramme de classe
 - b. Chaîne de Markov
 - c. *Réseau Bayésien*
 - d. Arbre de décision



13. Par quoi se définit un graphe ?
- a. Uniquement ses noeuds
 - b. Uniquement ses sommets
 - c. Les valeurs de ses relations et les noeuds qu'il possède
 - d. *Les nœuds, les sommets, les relations entre les nœuds et les valeurs de chaque relation*
14. Dans le cadre de la modélisation des graphes, qu'est-ce qui distingue un graphe orienté d'un graphe non orienté ?
- a. *Qu'il soit caractérisé par des arcs (graphe orienté) ou par des arêtes (graphe non orienté).*
 - b. Qu'à chaque relation soit obligatoirement associée une unique valeur (graphe orienté) ou non (graphe non orienté).
 - c. Leur nombre de noeuds
 - d. Leur nombre de sommets

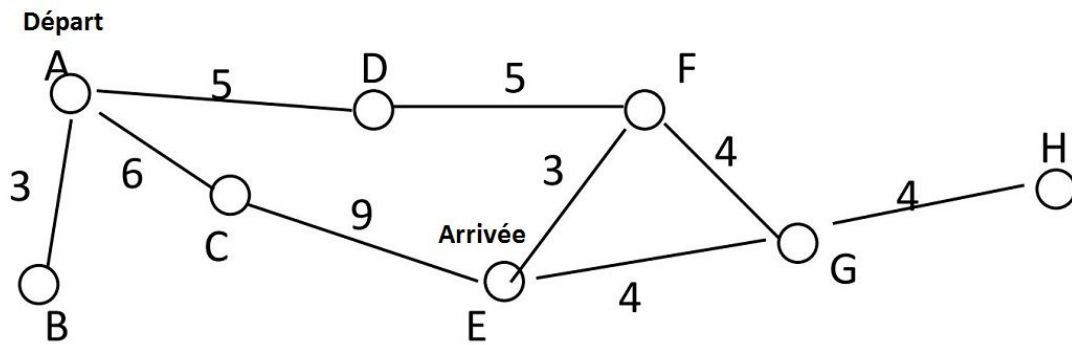
15. Combien de nœud(s) le graphe associé à cette matrice d'adjacence possède-t-il ?

0	1	0	0
1	1	1	1
0	1	0	1
0	1	1	0

- a. 2
 - b. 4**
 - c. 9
 - d. 16
16. Si un graphe n'est pas orienté, sa matrice d'adjacence est :
- a. Symétrique**
 - b. Inversible
 - c. Réflexive
 - d. Antisymétrique
17. Qu'est-ce qui est juste au sujet de l'algorithme de Dijkstra ?
- a. L'algorithme de Dijkstra permet de trouver le plus court chemin si on l'applique à un graphe dans lequel on a associé à certains arcs une valeur négative
 - b. C'est un algorithme d'évaluation de coup pour les jeux au tour par tour
 - c. L'algorithme explore intelligemment toutes les solutions et élimine au fur et à mesure celles qui sont aberrantes
 - d. L'algorithme explore toutes les solutions possibles puis choisit la meilleure**
18. Quelle est le point commun entre A* et Dijkstra ?
- a. L'utilisation d'une heuristique
 - b. Les états finaux
 - c. Ce sont des algorithmes de parcours de graphes**
 - d. Leur rapidité d'exécution
19. Qu'est ce qu'un coût heuristique ?
- a. Une notion qui apparaît dans l'algorithme A***
 - b. Le coût estimé entre le nœud courant et le nœud d'arrivée**
 - c. Le coût entre le nœud n et le nœud n+1
 - d. Une notion qui apparaît dans l'algorithme de Dijkstra
20. On souhaite parcourir un graphe et trouver le plus court chemin entre deux points donnés. Illustrer le chemin choisi, à l'aide de l'algorithme de Dijkstra, en précisant les points. Les 2 graphes qui peuvent être donnés sont en annexe 2.

Annexe 2 : GRAPHES EXERCICE DIJKSTRA

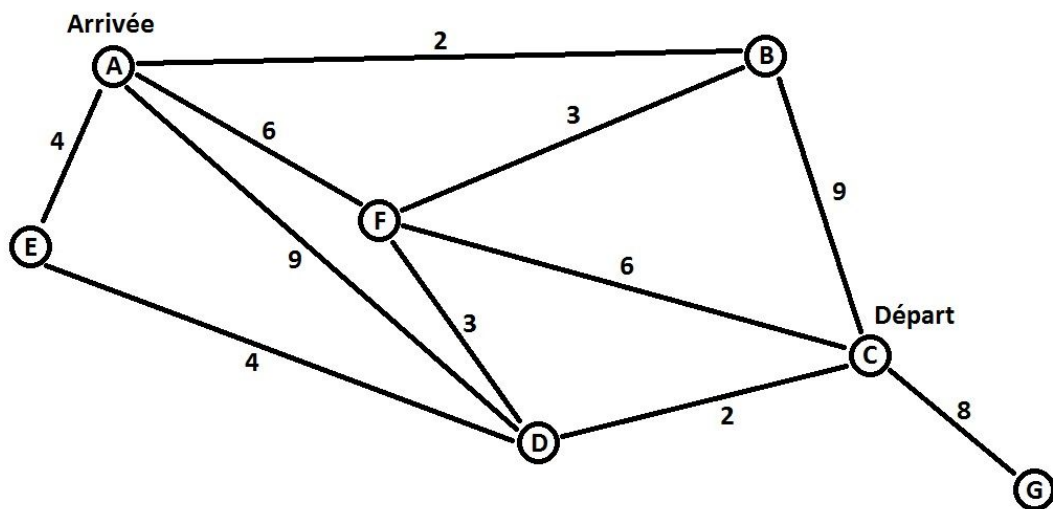
Graphe n°1 :



graphe0.txt

```
nombre de noeuds : 8
initial : 0
final : 6
arc1: 0 1 3
arc2: 0 2 6
arc3: 0 3 5
arc4: 2 4 9
arc5: 3 5 5
arc6: 4 6 4
arc7: 4 5 3
arc8: 6 7 4
arc9: 5 6 4
```

Graphe n°2 :



graphe1.txt

```
nombre de noeuds : 7
initial : 2
final : 0
arc1: 0 1 2
arc2: 1 2 9
arc3: 2 3 2
arc4: 3 4 4
arc5: 4 0 4
arc6: 3 0 9
arc7: 3 5 3
arc8: 2 5 6
arc9: 2 6 2
arc10: 3 5 6
arc11: 0 5 6
```

Annexe 3 : DIAGRAMME DE CLASSES

