

Logiciels et systèmes intelligents (LSI)
Département Informatique



Rapport de devoir 2
Méthodologies de l'IA S3

Développez une application Desktop Java

Réalisé par :
Mahjoubi Redwane
Ait-Abbou Samir
Outaleb Asmaa

Encadré par : MR.M'hamed AIT KBIR

Remerciement

C'est avec un réel plaisir que nous adressons les plus sincères remerciements à notre chère professeur **MR.M'hamed AIT KBIR** ,

pour leurs conseils précieux, leur soutien et leur compréhension à ses étudiants.

Merci infiniment

Table des matières

Remerciement	ii
Table des matières	iii
Table des figures	iv
Table des figures	iv
L'objectif	1
0.1 Architecture du Projet :	2
0.2 Choix d'un modeles se labyrinthes :	2
0.3 Les Composantes du Labyrainthes :	3
0.4 Gagner un Bonus :	4
0.5 traverser un obstacle :	5
0.6 demander de montrer le Goal :	6
0.7 demander de montrer le plus proche bonus :	9
0.8 Gérer le temps allouer pour le jeux :	10
0.9 Gérer le score :	11
Conclusion	12

Table des figures

1	Maze	1
2	Architecture du Projet	2
3	Choix d'un modeles se labyrinthes	2
4	Les Composantes du Labyrainthes	3
5	Gagner un Bonus	4
6	traverser un obstacle	5
7	Avant demande de l'aide	6
8	après demande de l'aide	7
9	Autre Situaion	8
10	demander de montrer le plus proche bonus	9
11	Temps expiré	10
12	Gérer le score	11

L'objectif

représenter un problème comme un graphe d'états en donnant une convenable représentation et mettre en application les algorithmes des stratégies de recherche vus dans le cours

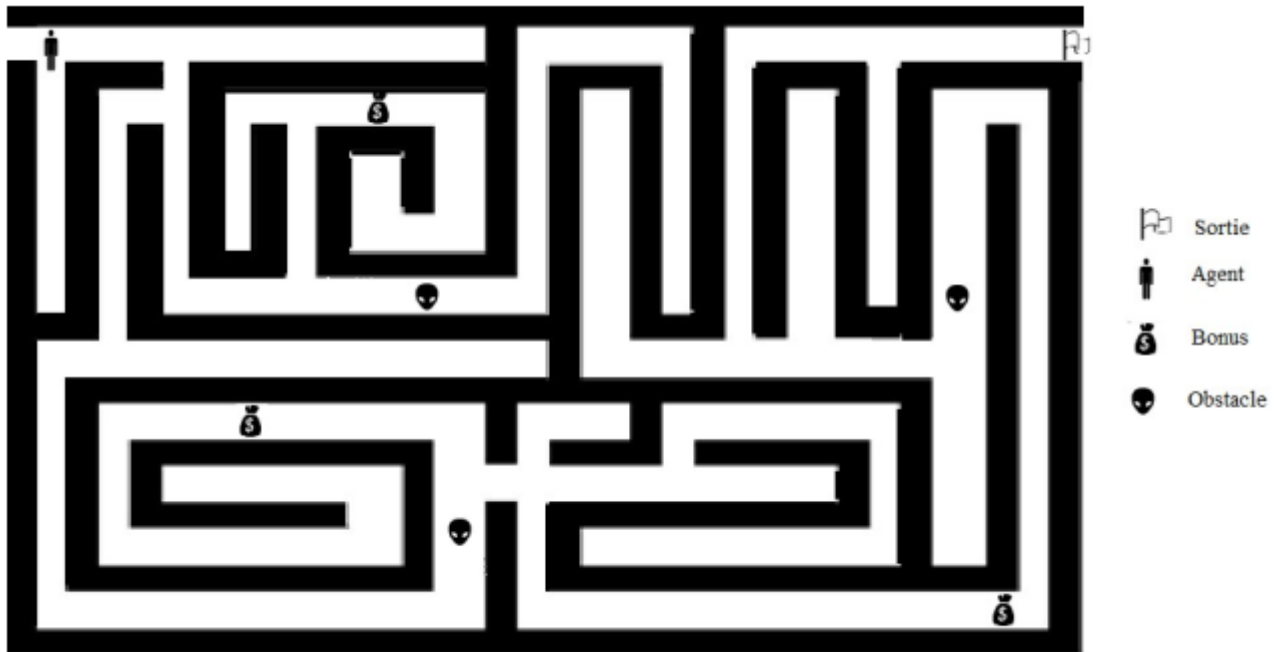


FIGURE 1 – Maze

Architecture du Projet

0.1 Architecture du Projet :

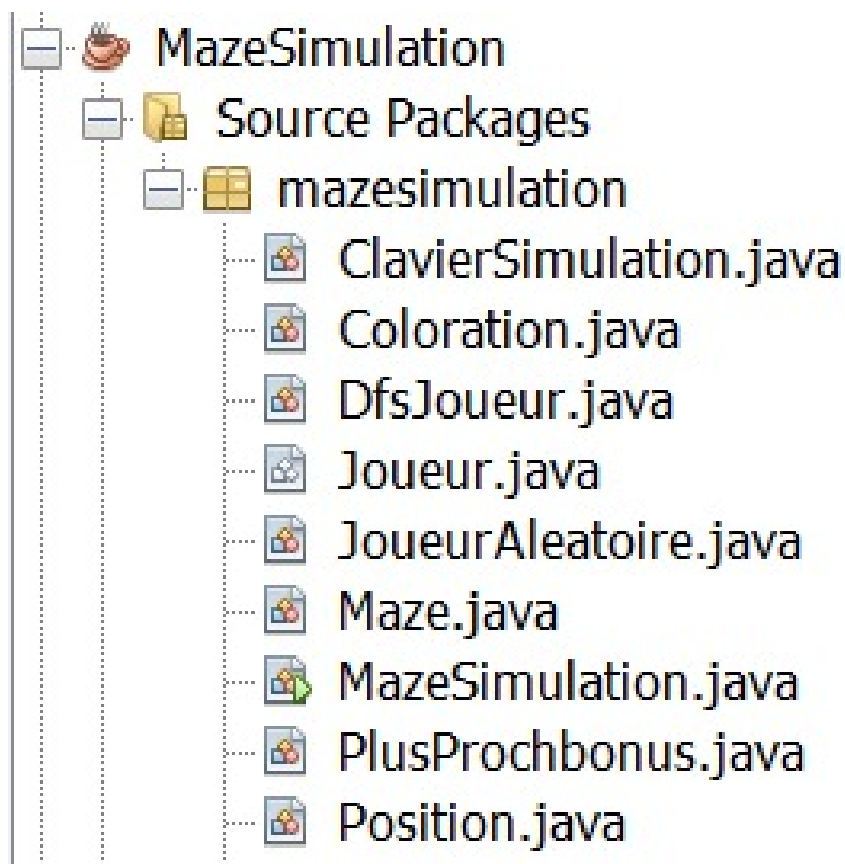


FIGURE 2 – Architecture du Projet

Choix d'un modeles se labyrinthes

0.2 Choix d'un modeles se labyrinthes :

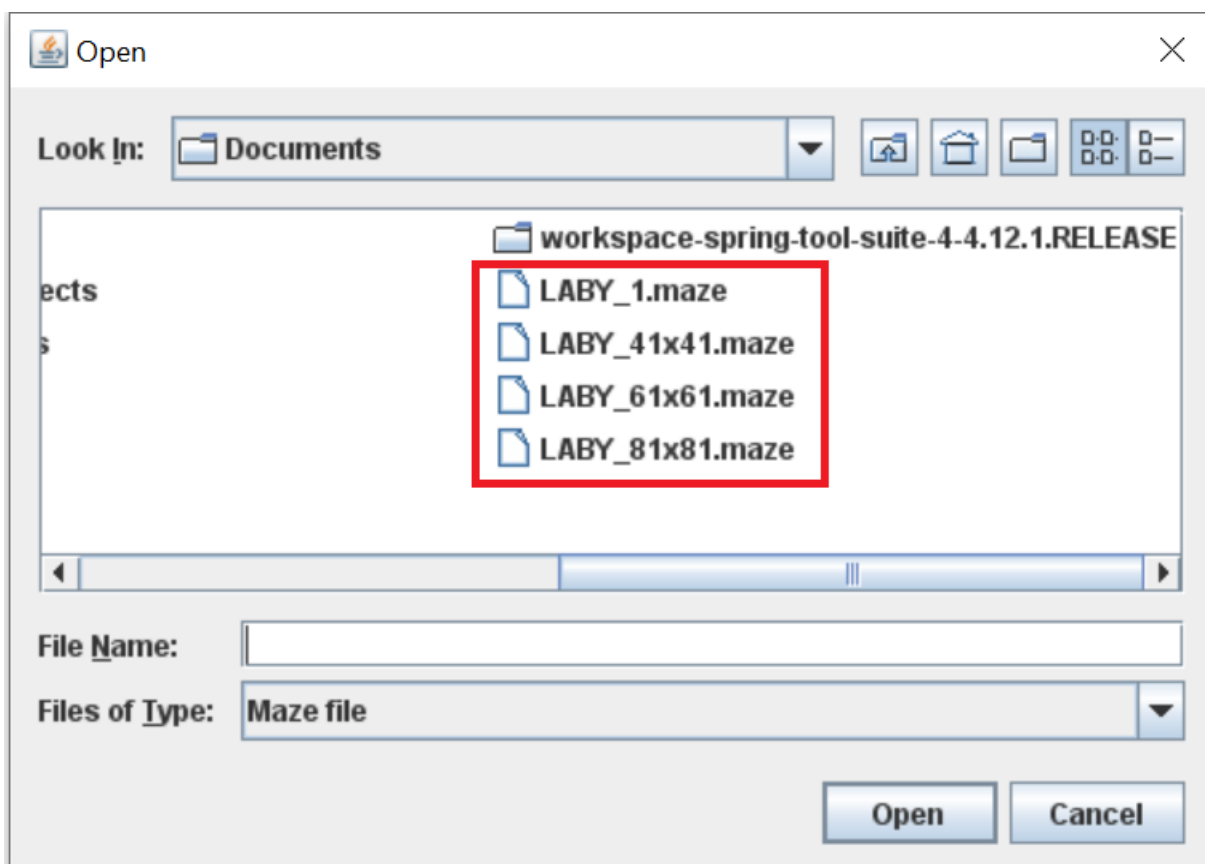


FIGURE 3 – Choix d'un modeles se labyrinthes

Les Composantes du Labyrinthes

0.3 Les Composantes du Labyrinthes :

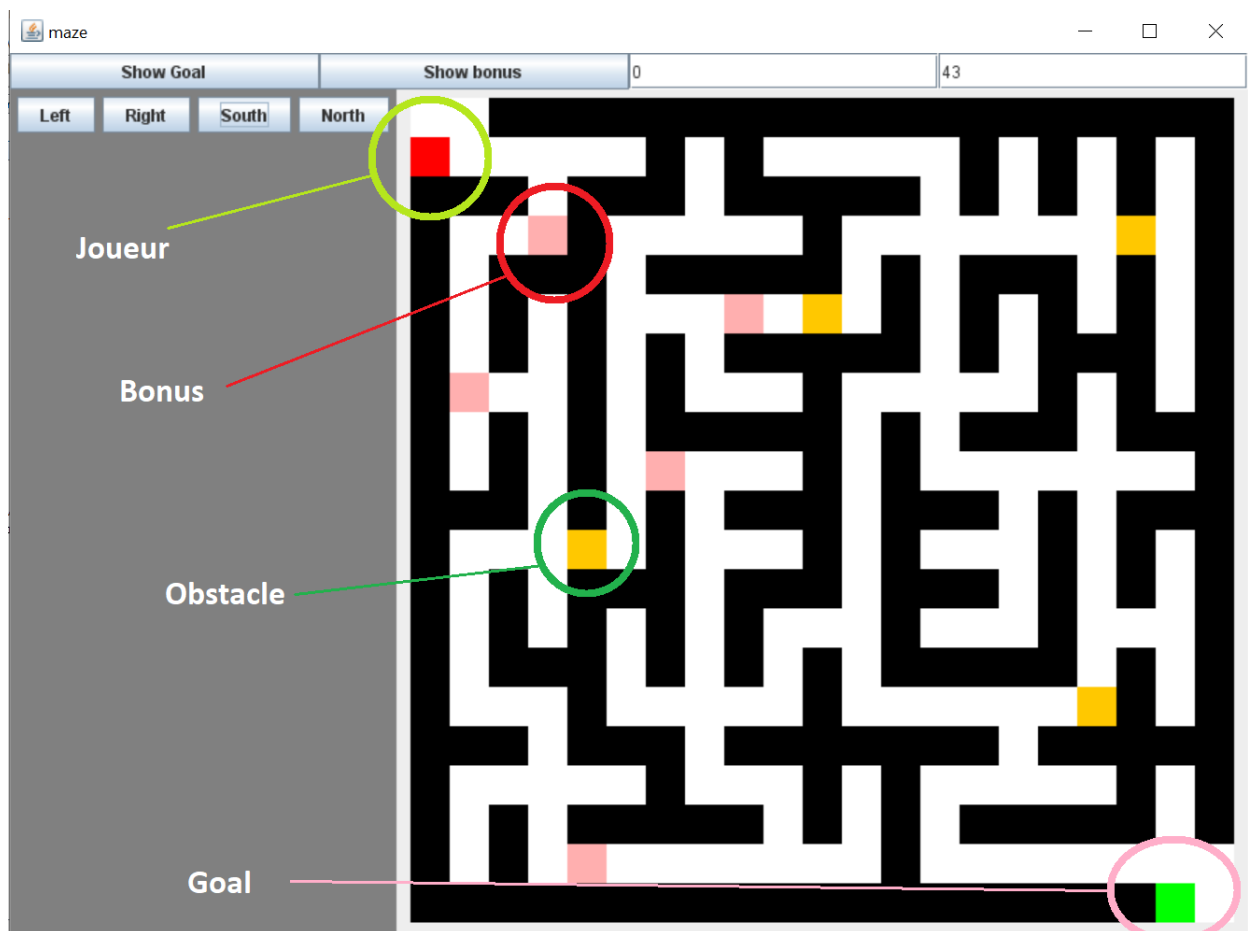


FIGURE 4 – Les Composantes du Labyrinthes

Gagner un Bonus

0.4 Gagner un Bonus :

Le joueur dispose d'un score nul au début qu'il peut alimenter en gagnant des bonus, 5 points par bonus.

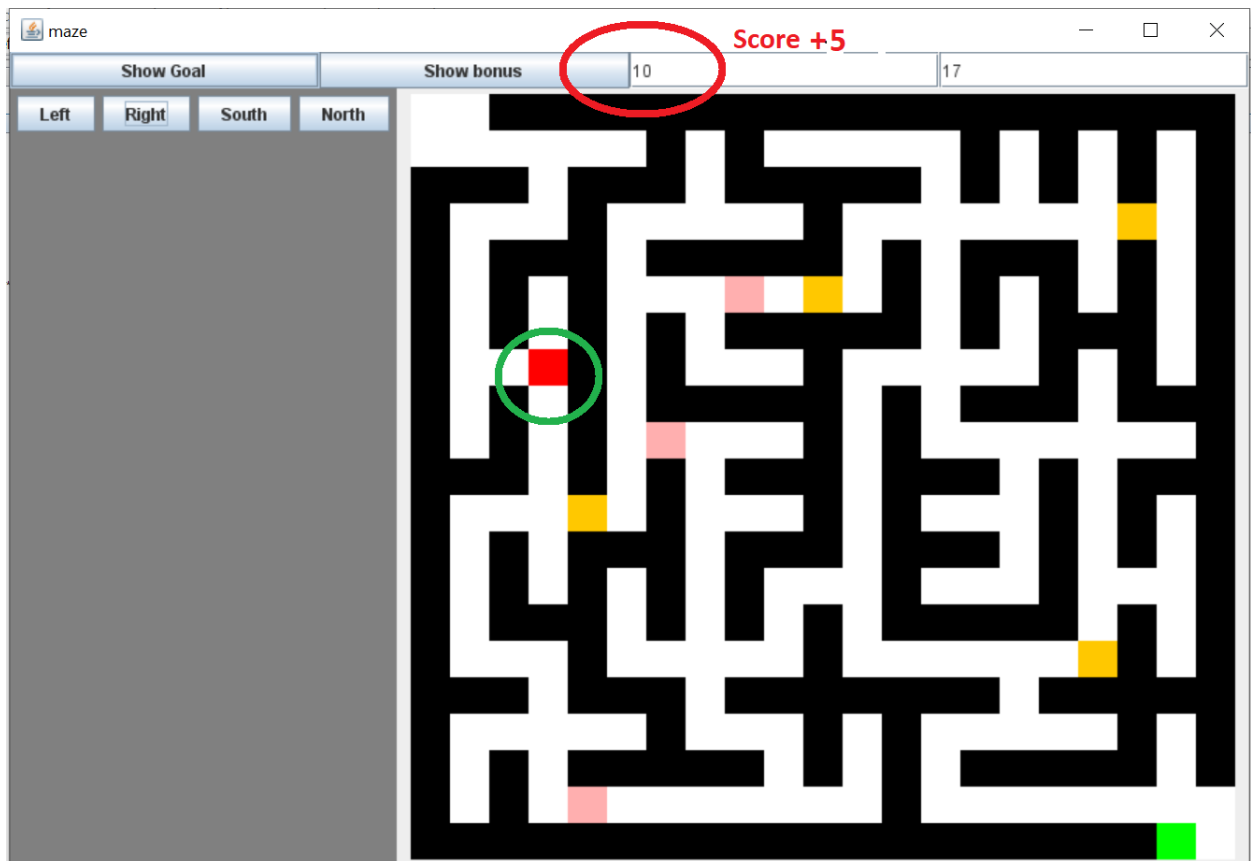


FIGURE 5 – Gagner un Bonus

traverser un obstacle

0.5 traverser un obstacle :

le joueur perd 2 points quant-il traverse un obstacle .

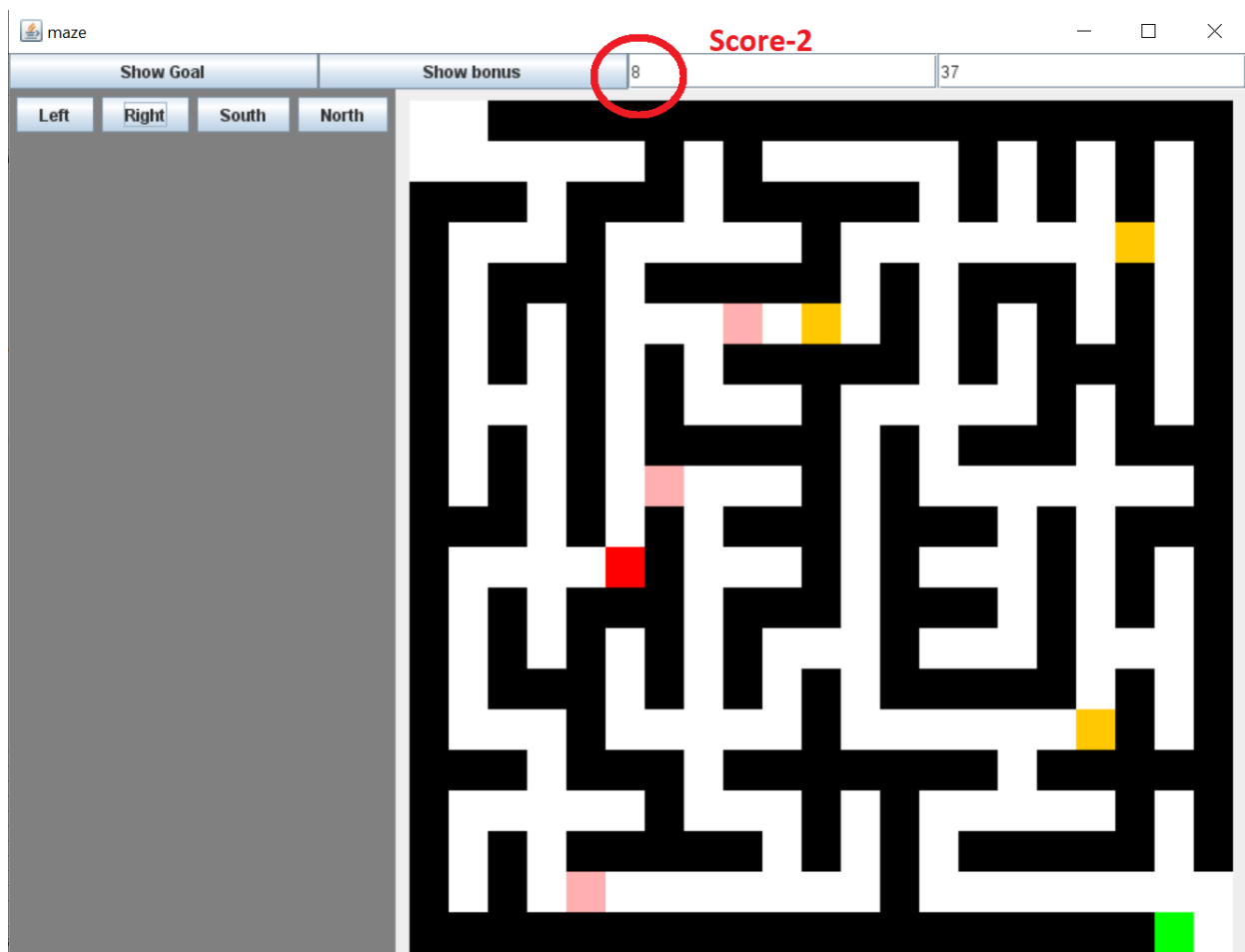


FIGURE 6 – traverser un obstacle

demander de montrer le Goal

0.6 demander de montrer le Goal :

le joueur perd 1 points lorsqu'il demande de l'aide pour visualiser temporairement un chemin pour atteindre la sortie

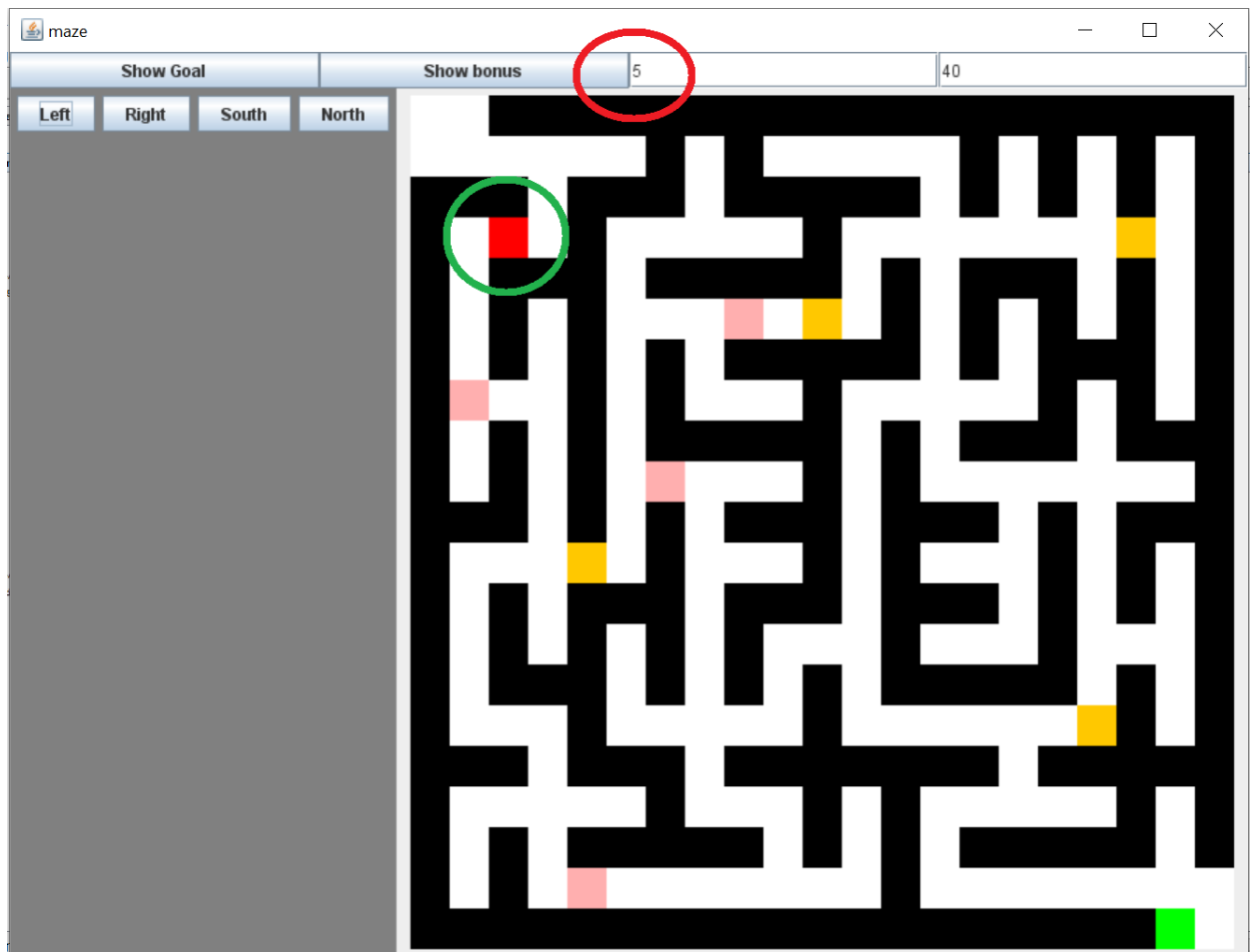


FIGURE 7 – Avant demande de l'aide

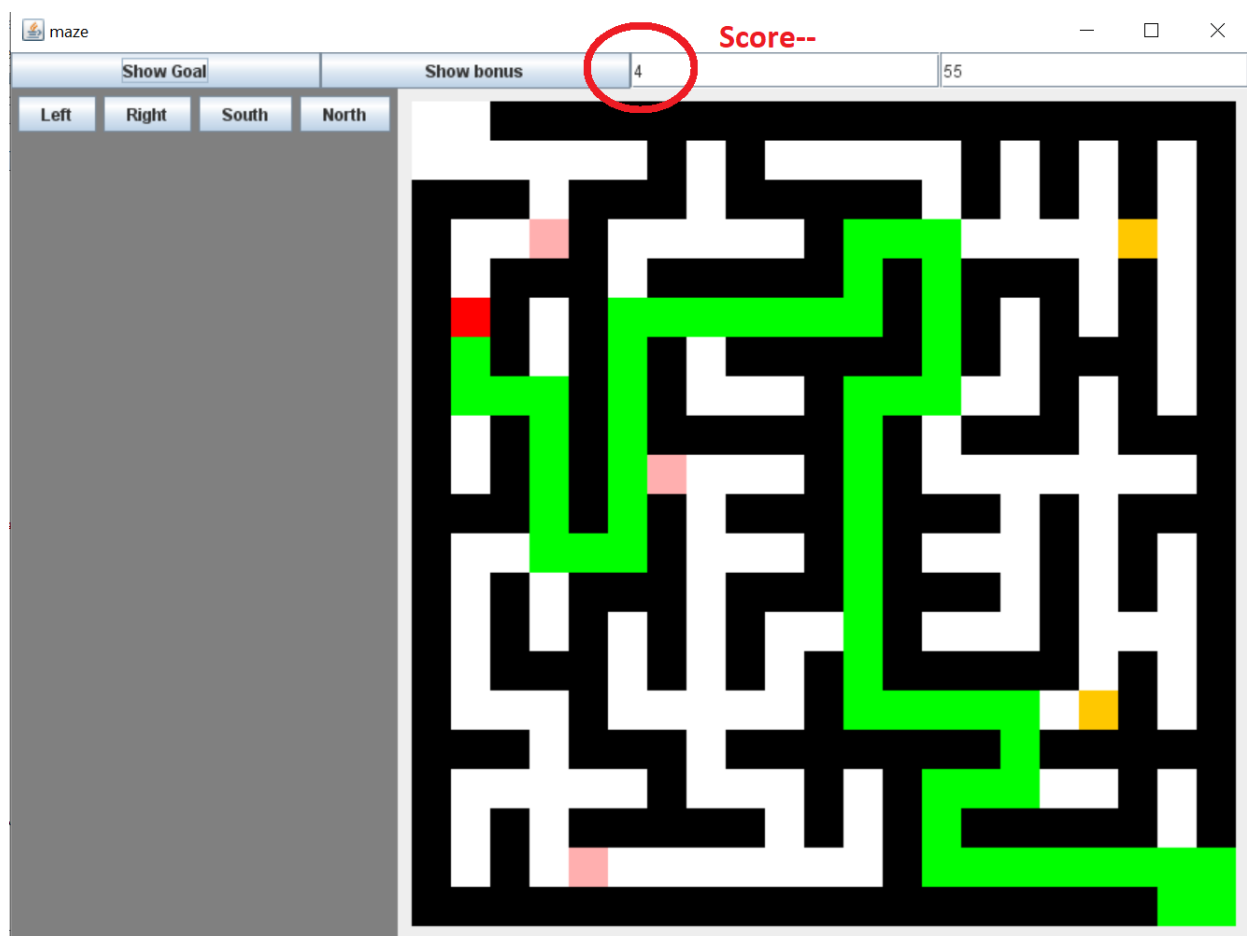


FIGURE 8 – après demande de l'aide

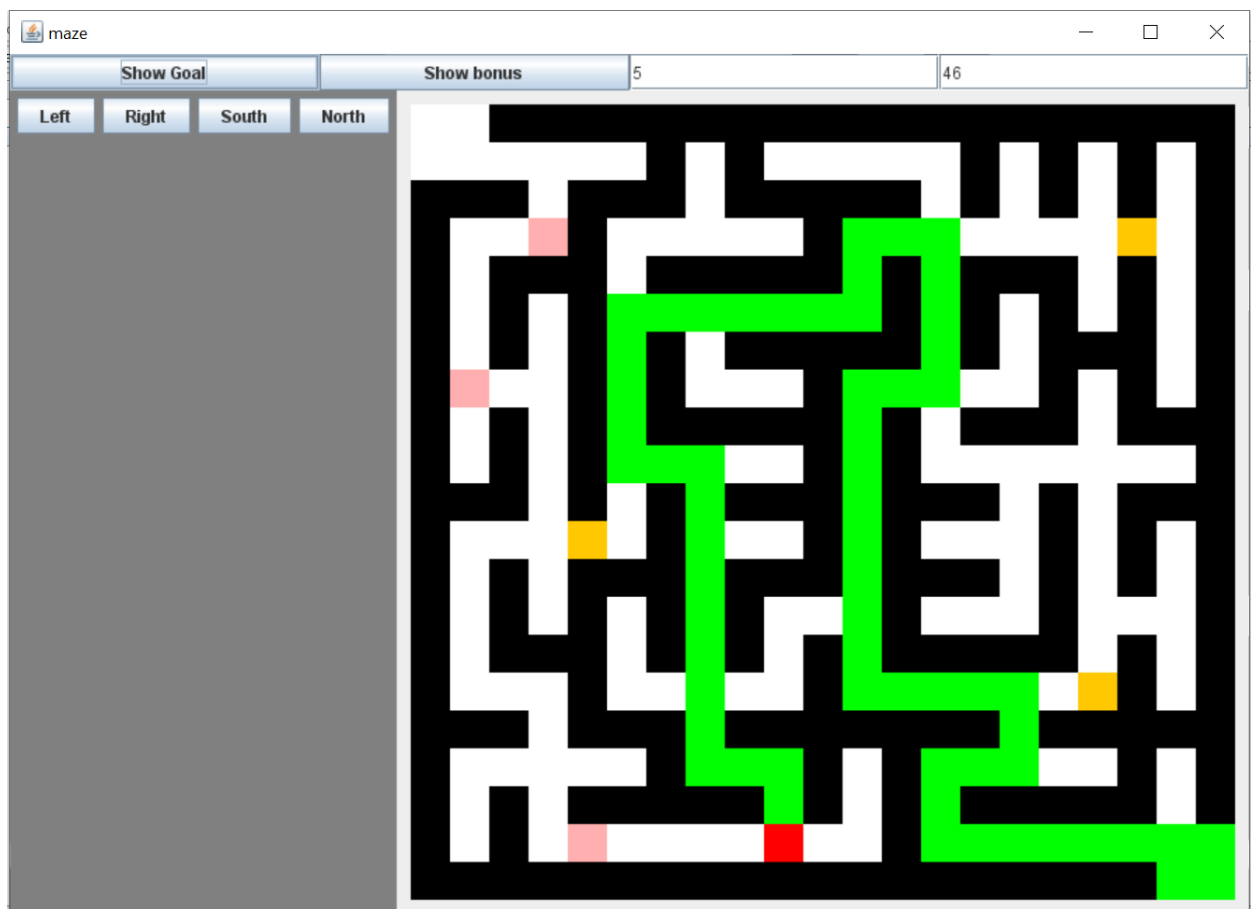


FIGURE 9 – Autre Situaion

demander de montrer le plus proche bonus

0.7 demander de montrer le plus proche bonus :

le joueur perd des points lorsqu'il demande de l'aide pour visualiser temporairement un chemin pour atteindre un bonus ($3 \times a$ points pour la sortie, a dépend de la stratégie à utiliser). Lors de la demande de d'aide, permettre le choix de la méthode de résolution : recherche en profondeur d'abord ($a=1/3$), recherche en largeur d'abord ($a=2/3$) ou A^* ($a=1$)

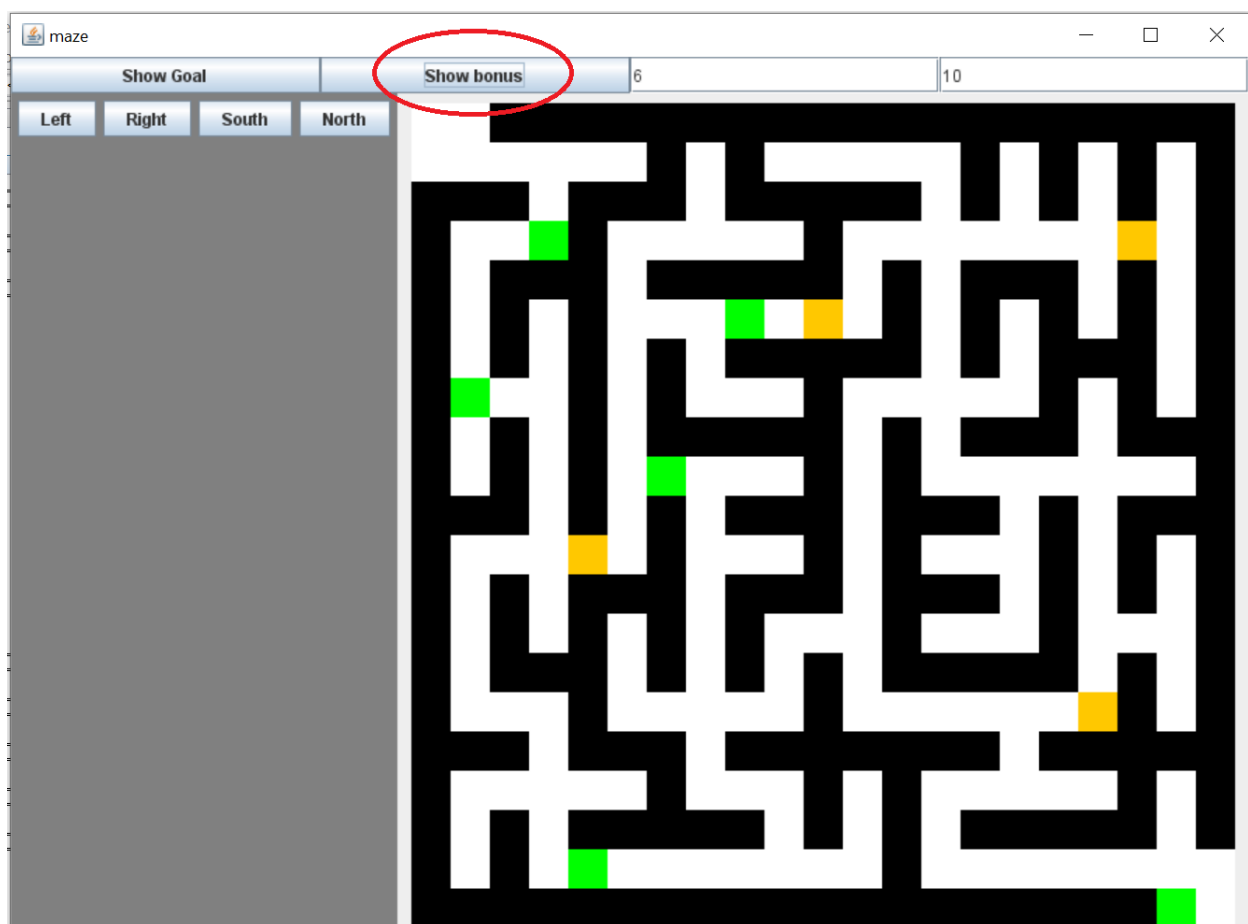


FIGURE 10 – demander de montrer le plus proche bonus

Gérer le temps allouer pour le jeux

0.8 Gérer le temps allouer pour le jeux :

le jeux s'arrete lorsque le temps s'expire

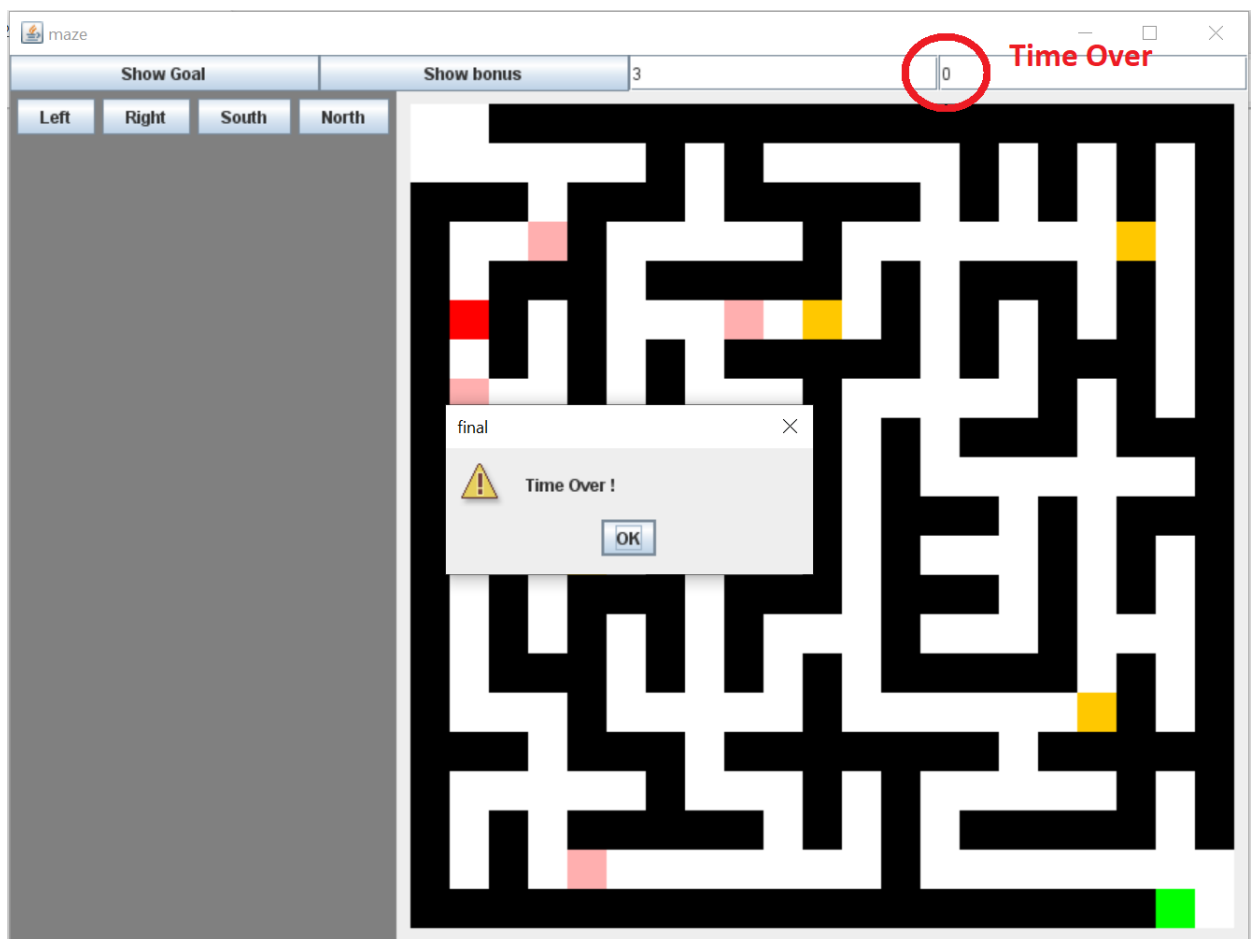


FIGURE 11 – Temps expiré

Gérer le score

0.9 Gérer le score :

le jeux s'arrete lorsque le joueur attient un score nul

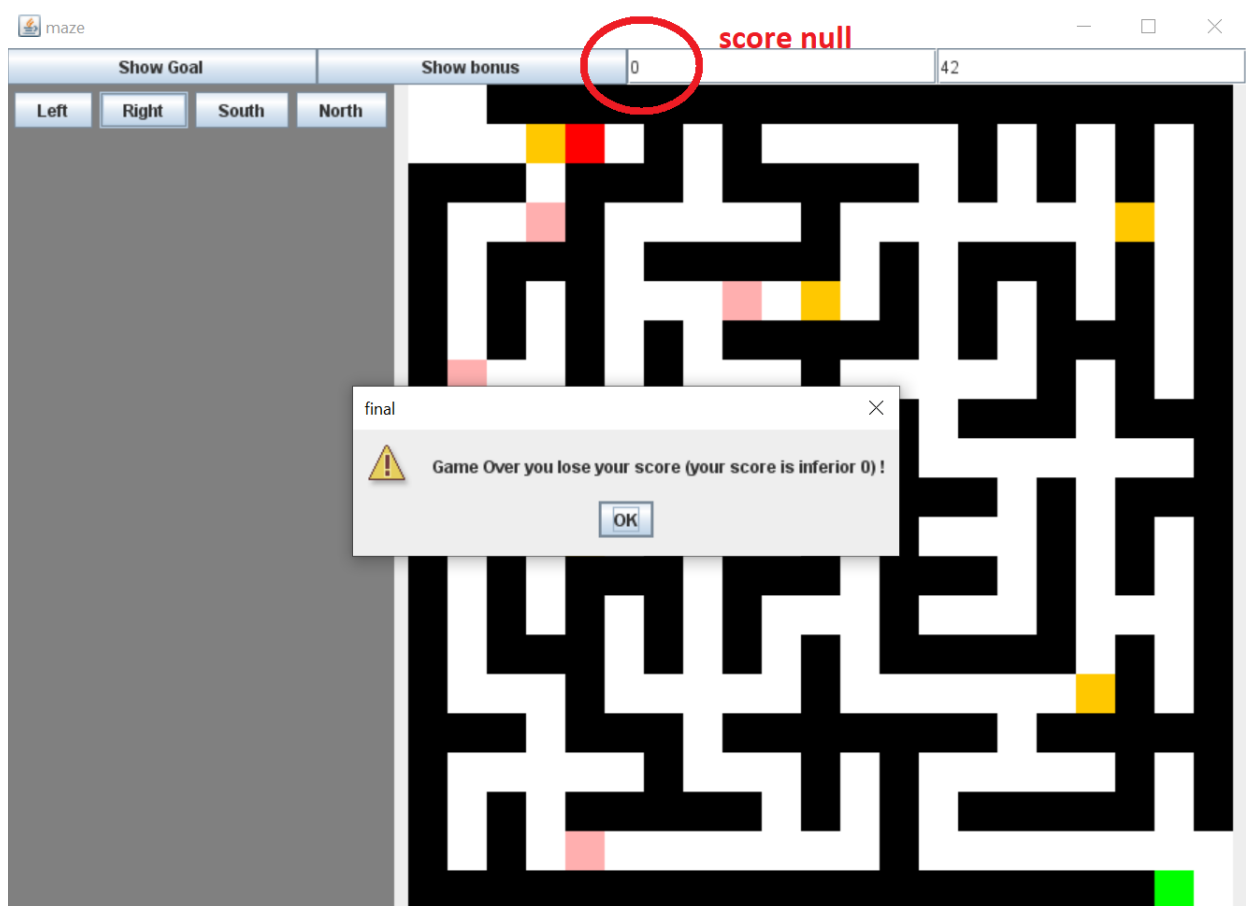


FIGURE 12 – Gérer le score

Conclusion

ce mini projet nous a permet d'implementer les algorithmes de l'intelligence artificiel (Dfs,Bfs,A*) et de bien se familiariser avec java swing .