CASATI Pierre - IG4

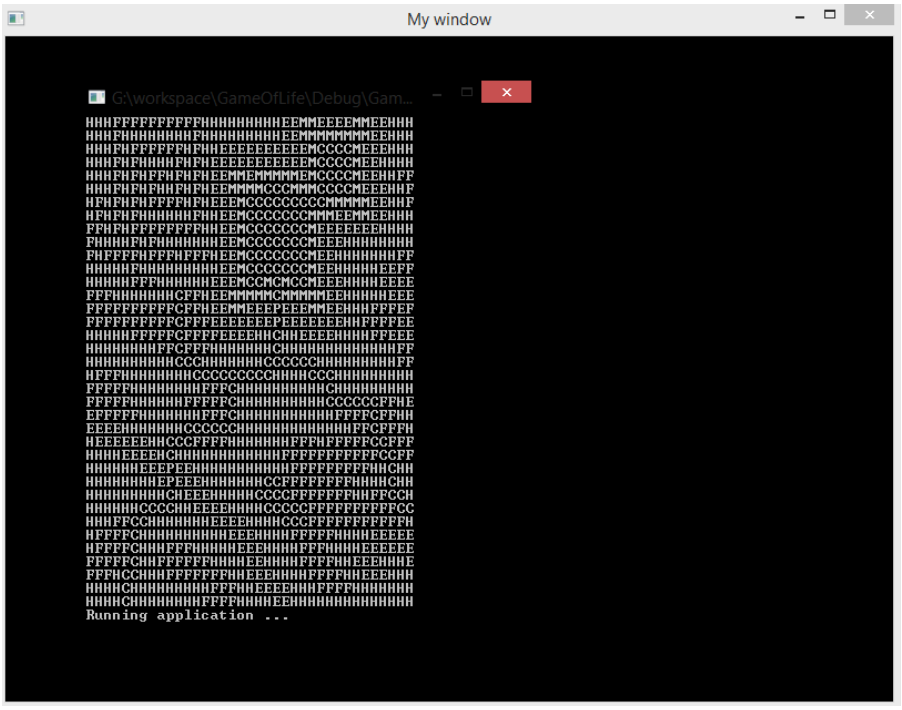
Projet Graphe : Algorithme A*

Introduction :

Au début du projet, j'ai choisi de développer en C++, pour deux raisons.
Premièrement, le choix étant libre, et comme je suis passionné par le C++, j'avais enfin l'occasion de le pratiquer pour un projet.
Deuxièmement, j'avais déjà commencé un projet qui s'accordait parfaitement avec le thème du projet de Graphe.

Présentation du Projet C++

Le but initial de mon programme était d'implémenter un System Entity Component Pattern, une architecture logiciel très souvent utilisée dans le développement de moteur de jeux vidéos. Le but de l'ECS est de pallier au problème des arbres d'héritages démesurés. Au lieu de s'axer sur les services que rendent chaque classe, la structure du programme est axé sur les données. Pour chaque « entité », des « composants » sont générés de manière indépendante et sont ensuite traités par l'intermédiaire de « systèmes ». Cette structure à l'avantage d'être plus souple et plus performante.
Cependant, mon architecture se concentrait essentiellement sur la génération d'une carte composée de cases (ou de cube). C'est là que j'ai fais le lien avec les graphes !
J'ai donc choisi de continuer de développer mon système et d'implémenter l'algorithme A* directement dedans.



Dans le cadre du projet, le programme se décomposait ainsi : les entités cases étaient associées à des

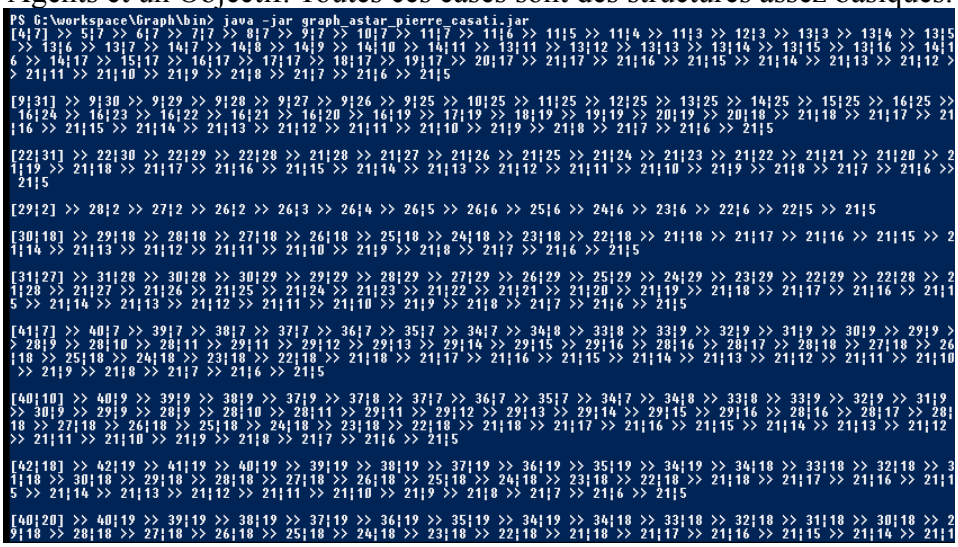
composants « Coordonnée », « Texture », « Temps de traversée », « Coût », « Voisin », etc …
Un système se chargeait du chargement de la carte depuis un fichier Bitmap, un autre exécutait l'algorithme A*, et un 3ème générait le rendu.

Problème rencontré :

La partie affichage console était fonctionnelle et je m'apprêtait à faire un rendu en fenêtre Windows. De plus, l'algorithme A* était également fonctionnel. Cependant, je me suis aperçu, que je n'avais pas de moyen de compiler mon programme pour Linux, l'OS cible. En effet, je développe sous C++14, une norme récente, et les compilateurs à jours ne sont pas encore très fréquents. (Le mien a même été compilé à la main !).

Solution Finale :

J'ai donc opté à la dernière minute pour un changement radical : j'ai réimplémenté l'algorithme en Java (langage portable) avec un affichage console.
Le code est énormément simplifié en comparaison du programme C++. Il y a un tableau de Cases, ainsi que des Agents et un Objectif. Toutes ces cases sont des structures assez basiques.



Spécification fonctionnelle :

- A partir d'un tableau, lire la carte du problème posé

- Générer le graphe correspondant

- Exécuter l'algorithme A* pour chaque agent présent

- Retourner et afficher le(s) chemin(s)

Résultat :

Chaque paragraphe affiché correspond à la solution d'un agent, décrit de la manière suivante :

[départX |départY] >> étape1X | étape1Y >> …. >> objectifX | objectifY