

TP3

Recherche de chemins (*Pathfinding*)

Auteurs : Vincent Echelard
Kyle Ross

Table des matières

A.	MISE EN CONTEXTE	2
B.	DESCRIPTION SOMMAIRE DU TRAVAIL PRATIQUE	2
C.	DESCRIPTION DU PROJET DE BASE.....	3
1.	<i>Introduction</i>	3
2.	<i>Classe Carte</i>	3
3.	<i>Struct Point2D</i>	3
D.	DESCRIPTION DES TÂCHES À RÉALISER	4
1.	<i>Introduction</i>	4
2.	<i>Résultats d'exécution attendus</i>	5
E.	CONDITIONS DE RÉALISATION ET DE REMISE	6

420-203-RE
Développement de programmes
dans un environnement graphique

A. *Mise en contexte*

Ce travail vous permettra principalement de vous familiariser avec la programmation, des algorithmes de parcours de graphes en général, et des algorithmes de recherche de chemins en particulier.

B. *Description sommaire du travail pratique*

Les BBB (Big Brain Bangers) ont revu le design de leur projet de développement de jeu et prévoient maintenant ajouter un volet terrestre à leur jeu de combat spatial. Pour conquérir des systèmes solaires, les joueurs devront poser leur vaisseau sur des planètes et combattre les ennemis qui s’y trouvent. Comme nous l’avons vu dans le dernier TP, le Game Designer des BBB manque désespérément d’originalité, et il a choisi de s’inspirer « fortement » de grands classiques du genre (*StarCraft*, *Command and Conquer* et autres RTS) dans la conception de ce nouveau projet.

Avant d’aller, encore une fois, demander de l’argent à l’éditeur de jeux EA (*Evil Association*) pour être capables de développer ce nouveau volet du jeu, ils doivent de nouveau démontrer leur savoir-faire. Étant donné que personne ne vous surpasse en programmation dans l’entreprise, vous avez hérité de la tâche de concevoir les algorithmes recherche de chemins qui vous permettront de mener vos troupes à l’assaut des défenses ennemies.

Vous devrez donc concevoir au minimum deux algorithmes :

1. un algorithme d’exploration en largeur (*Breadth-First Search*) ; et,
2. un algorithme adaptatif, soit celui de Dijkstra.

Évidemment, si vous souhaitez réellement impressionner votre éditeur de jeu et, par le fait même, obtenir quelques « dollars » (des points) supplémentaires, vous devrez alors développer en plus un algorithme adaptatif et informé, soit A*.

C. Description du projet de base

1. Introduction

Votre prototype prendra la forme d'un projet C# standard de type console. Vous devrez compléter le travail préparé par vos professeurs (vos profs). Vous recevrez donc un projet à compléter contenant quelques composants logiciels qui pourraient vous être utiles.

2. Classe Carte

Cette classe permet de lire un fichier texte contenant la description d'une carte et permet d'obtenir un modèle simplifié d'accessibilité qui se résume à être un tableau de `bool` où Vrai signifie « accessible » et Faux « inaccessible ». Vous pouvez accéder aux données de ce tableau par l'intermédiaire d'une propriété indexée.

De plus, cette classe met à votre disposition quatre autres propriétés :

1. une propriété qui vous permet d'obtenir la **largeur** de la carte;
2. une propriété qui vous permet d'obtenir la **hauteur** de la carte;
3. une propriété qui vous permet d'obtenir la position de la **source** sur la carte; et,
4. une propriété qui vous permet d'obtenir la position de la **destination** sur la carte.

Cette classe contient une constante représentant le chemin d'accès vers le fichier contenant la carte, il est possible que vous ayez à modifier cette constante, si la structure de répertoires de votre projet diffère de la mienne.

3. Struct Point2D

Cette struct permet de représenter une simple coordonnée 2D composée de deux valeurs entières. Ce composant logiciel est déjà utilisé dans la classe Carte, mais il est fort probable que vous ayez besoin de l'utiliser dans vos algorithmes.

D. Description des tâches à réaliser

1. Introduction

Vous devez concevoir quelques composants logiciels qui vous permettront d’implémenter les algorithmes de recherche de chemins demandés. Nous vous proposons tout d’abord de faire une lecture attentive du site suivant :

<https://www.redblobgames.com/pathfinding/a-star/introduction.html>

Ce site vous propose une vision simplifiée du développement des algorithmes de recherche de chemin qui est plutôt axée sur la compréhension que sur la performance. Vous y trouverez tout ce qu’il faut pour la conception des deux algorithmes obligatoires, ainsi que pour la conception de l’algorithme demandé en bonus.

Naturellement, vous pouvez également faire d’autres choix en ce qui concerne votre stratégie de développement. L’important est de viser la compréhension des algorithmes et des structures (listes, files et piles) que vous manipulerez. Ne faites surtout pas l’erreur de copier un programme sur un site quelconque, cela ne vous mènera pas très loin. Tout d’abord, il est fort probable que nous trouvions sans difficulté le site de provenance de votre code (concept de plagiat...). Ensuite, cela vous causera des problèmes lorsque vous devrez éventuellement concevoir un algorithme adapté pour une PFI par exemple.

E. Conditions de réalisation et de remise

- a) Ce travail **doit** être réalisé en équipe de deux.
- b) Vous devrez remettre, par **Colnet**, un fichier d'archives contenant votre projet C# que vous aurez pris soin de nettoyer (retirer le répertoire BIN).
- c) Vous devez nommer ce fichier d'archives (ZIP) en respectant les règles de toponymie suivante : « **TP3 – Nom1 Prénom1 – Nom2 Prénom2** ».
- d) Vous devrez remettre le fichier d'archives par l'intermédiaire de la boîte de remise associée à l'évaluation.
- e) Ce fichier d'archives devra être remis d'ici minuit le **mercredi 25 novembre** prochain.