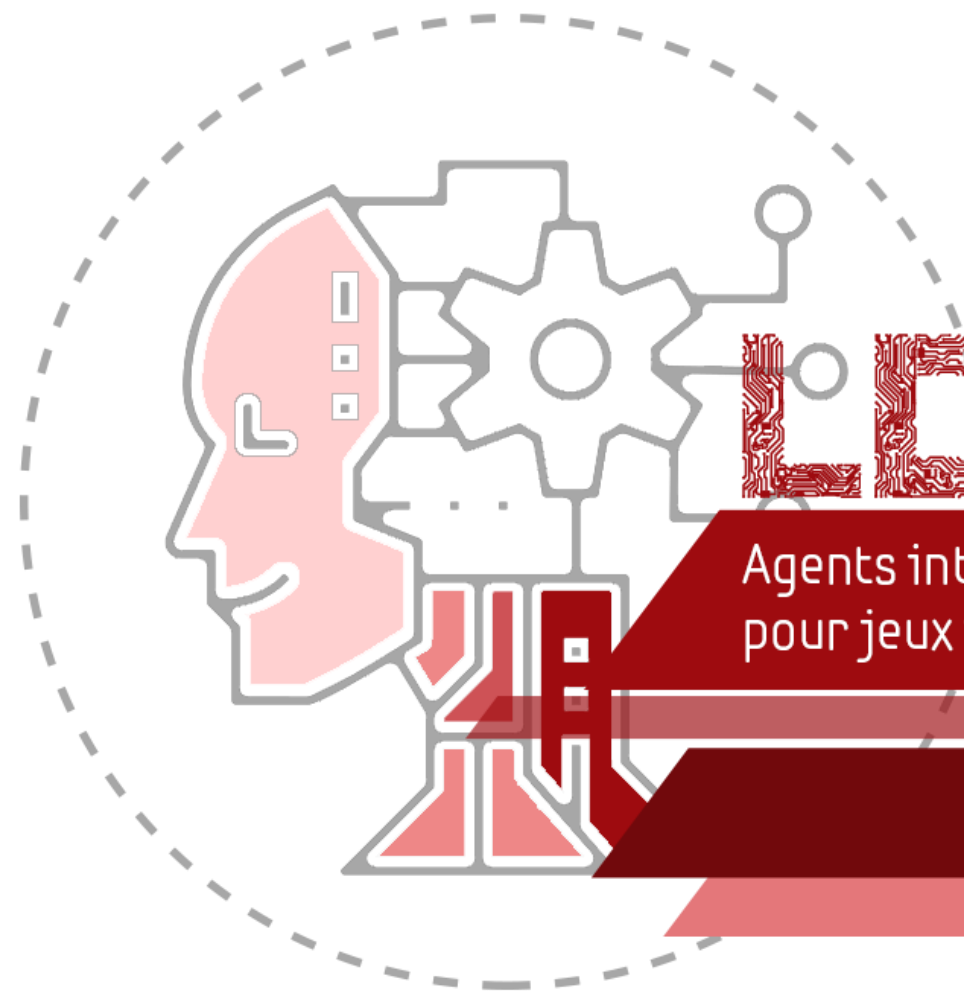


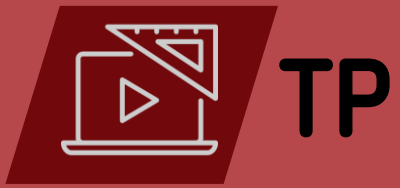


POLYTECHNIQUE
MONTRÉAL



LOG 8235

Agents intelligents
pour jeux vidéo



TRAVAUX PRATIQUE 2

Introduction

Introduction

- Le but de ces travaux pratiques est de mettre en application les différentes notions abordées lors des divers modules de cours théorique.
- Les travaux doivent être réalisés à l'aide du cadre fourni basé sur le Unreal Engine. Le fil conducteur est le développement d'agents intelligents pour un jeu de style Pac-Man.
- Un certain nombre d'ingrédients vous sont fournis afin de vous concentrer pleinement sur les différentes tâches demandées (cf section Éléments Fournis).
- L'objectif de ce second TP est le développement l'amélioration des comportements du premier TP. Les différentes tâches demandées demanderont de mettre en pratique les techniques avancées de navigation ainsi que les techniques d'animation vu dans la partie théorique.



Travail à accomplir

Travail à accomplir

1. Génération du navmesh

- Un navmesh doit pouvoir être généré dans le niveau des travaux pratiques. La génération doit prendre en compte les différents éléments gameplay du niveau qui peuvent impacter le déplacement dont les dalles de type « Death ». Le navmesh, par le paramétrage de sa génération, doit être cohérent avec l'architecture et praticable par les agents et le joueur.

Éléments obligatoires	Points
Génération cohérente et praticable du navmesh	2 pts
Prise en compte des dalles de type « Death » dans le navmesh généré	3 pts

Travail à accomplir

2. Calcul et affichage d'un chemin

- L'agent doit pouvoir calculer un chemin jusqu'au collectible le plus proche. Le chemin ainsi calculé devra être affiché en utilisant les fonctions de debug d'Unreal. On peut considérer pour ce point que l'agent connaît la position des collectibles.
- Le calcul du chemin une fois la cible déterminée peut être fait directement ou par des fonctions existantes dans Unreal.

Éléments obligatoires	Points
Calcul d'un chemin vers le collectible le plus proche	3 pts
Affichage du chemin	2 pts

Travail à accomplir

3. Parcours d'un chemin

- L'agent doit pouvoir parcourir le chemin calculé vers le collectible le plus proche. (Aller voir le fichier `USDTPathFollowingComponent.cpp`)
- Il faut redéfinir dans ce component entre autre deux méthodes qui permettent le suivi de chemin (`SetMoveSegment`, `FollowPathSegment`).
- L'agent doit pouvoir suivre un chemin normal, sans saut dans un premier temps.

Éléments obligatoires	Points
Parcours fluide des différents points du chemin	5 pts

Travail à accomplir

4. Rajout de navlink

- La génération du navmesh doit comporter des navlinks qui permettent d'offrir des possibilités de chemins pour que l'agent puisse sauter par-dessus les dalles de type « Death ».

Éléments obligatoires	Points
Rajout de navlinks dans le navmesh généré	4 pts
Cohérence des navlinks	1 pt

Travail à accomplir

5. Parcours d'un chemin comportant des navlinks

- L'agent doit pouvoir ensuite parcourir un chemin dont certains points nécessitent que l'agent saute afin d'éviter un obstacle.
- Il faut maintenant adapter le suivi de chemin afin d'obtenir le comportement voulu au niveau des navlinks.

Éléments obligatoires	Points
Saut fluide de l'agent au-dessus des dalles de type « death »	3 pts
Trajectoire et vitesse de saut adaptée	2 pts

Travail à accomplir

6. Mis à jour du comportement de poursuite

- L'agent doit pouvoir calculer le chemin vers le joueur dans le cas où celui-ci est visible par l'agent et non boosté. Lorsque le joueur n'est plus visible, l'agent doit se diriger vers la dernière position où le joueur a pu être aperçu par l'agent (LKP).
- La poursuite est prioritaire sur le ramassage de collectibles.

Éléments obligatoires	Points
Détection et activation cohérente du comportement de poursuite	1 pt
Poursuite du joueur en parcourant un chemin vers lui quand il est visible	2 pts
Poursuite en parcourant un chemin vers la LKP sinon	2 pts

Travail à accomplir

7. Mis à jour du comportement de fuite

- L'agent doit adopter un comportement de fuite s'il détecte le joueur boosté
- L'agent doit pouvoir calculer un chemin l'éloignant du joueur en se dirigeant vers le point de fuite le plus pertinent dans le cas où son comportement de fuite est activé.

Éléments obligatoires	Points
Détection et activation cohérente du comportement de fuite	1 pt
Éloignement du joueur en parcourant le chemin vers le point de fuite le plus pertinent	4 pts

Travail à accomplir

8. Ajout d'animation de déplacement

- En utilisant les animations fournis, l'agent doit pouvoir se déplacer en supportant les 2 vitesses (marche, course).

Éléments obligatoires	Points
Graphe d'animation	2 pts
Support des vitesses de marche et course dans le graphe	3 pts

Travail à accomplir

9. Ajout d'animation de saut

- En utilisant les animations fournis, l'agent doit pouvoir utiliser une animation lorsqu'il doit sauter en utilisant un navlink.

Éléments obligatoires	Points
Graphe d'animation supportant le saut	5 pts



Éléments fournis



Éléments fournis 1/3

Blueprints Unreal

BP_DeathFloor

- Ingredient gameplay de type obstacle dont la fonction est de provoquer la téléportation au point de départ de l'entité ayant collisionné avec
- StaticMeshActor avec une collision de type "DeathObject"

BP_SDTCollectible

- Ingredient gameplay de type pickup dont la fonction est de disparaître lorsqu'une entité rentre en contact avec. Si l'entité est le joueur, le pickup confère alors l'état « PowerUp » pendant un certain temps au joueur

BP_SDTACharacter

- Classe de base devant être utilisé pour implémenter le code du pawn de l'agent

BP_SDTAController

- Classe de base devant être utilisé pour implémenter le code du controller de l'agent

BP_SDTMainCharacter

- Entité joueur qui doit être contrôlé par un humain.
- Utilise une collision de type « Player »

Éléments fournis 2/3

BP_SDTFleeLocation

- Entité placée dans le monde servant de point de repère quand un agent est en fuite

JumpCurve

- Courbe servant à spécifier la trajectoire de saut des agents

SDT_AnimBP

- Blueprint d'animation contenant la logique de sélection d'animation pour le mouvement de l'agent

Document

ToolBox-Polytechnique

- Liste de fonctions Unreal documentées à utiliser pour la réalisation du TP

Éléments fournis 3/3

C++

SDTAIController.h

- Contrôleur servant à contrôler les IAs. C'est la classe que vous allez principalement modifier pour implémenter votre IA »

SDTBaseAIController.h

- Contrôleur de base de l'IA SDTAIController.h hérite de celui-ci.

SDTPathFollowingComponent

- Composante qui implémente la logique de suivis de chemin. (À compléter).

SDUtils

- Contient des fonctions utilitaires ainsi que la définition de différents canaux de collision



évaluation

Évaluation 1/2

Travail à accomplir	Env. fourni
1. Génération du navmesh	5 pts
2. Calcul et affichage d'un chemin	5 pts
3. Parcours d'un chemin normal	5 pts
4. Rajout de navlinks	5 pts
5. Parcours d'un chemin comportant des navlinks de saut	5 pts
6. Mis à jour du comportement de poursuite	5 pts
7. Mis à jour du comportement de fuite	5 pts
8. Animation de déplacement	5 pts
9. Animation de saut	5 pts
Aspect global de la réalisation*	5 pts
Soin apporté au code**	-10 pts

Total : 50 points

Évaluation 2/2

*L'aspect global de la réalisation prendra en compte le soin apporté à la crédibilité des comportements de l'agent. Il sera apprécié entre autres :

- Crédibilité du déplacement de l'agent
- Qualité d'enchaînement des animations et des comportements

**Le soin apporté au code appréciera les éléments suivants :

- Taille des fonctions
- Factorisation du code
- Lisibilité et maintenabilité



Remise

Remise

Pour la remise, veuillez me remettre un rapport pdf.

Ce rapport devra contenir les informations suivantes :

Vos noms et matricules, ainsi que pour chaque question une liste des fichiers et fonctions vous avez modifié et des explications si nécessaire.

La solution devra être un fichier .zip contenant les fichiers et répertoires suivants :

Config/...

Content/... (Sauf StarterContent et si le fichier est trop lourd, seulement les éléments modifiés)

Source/...

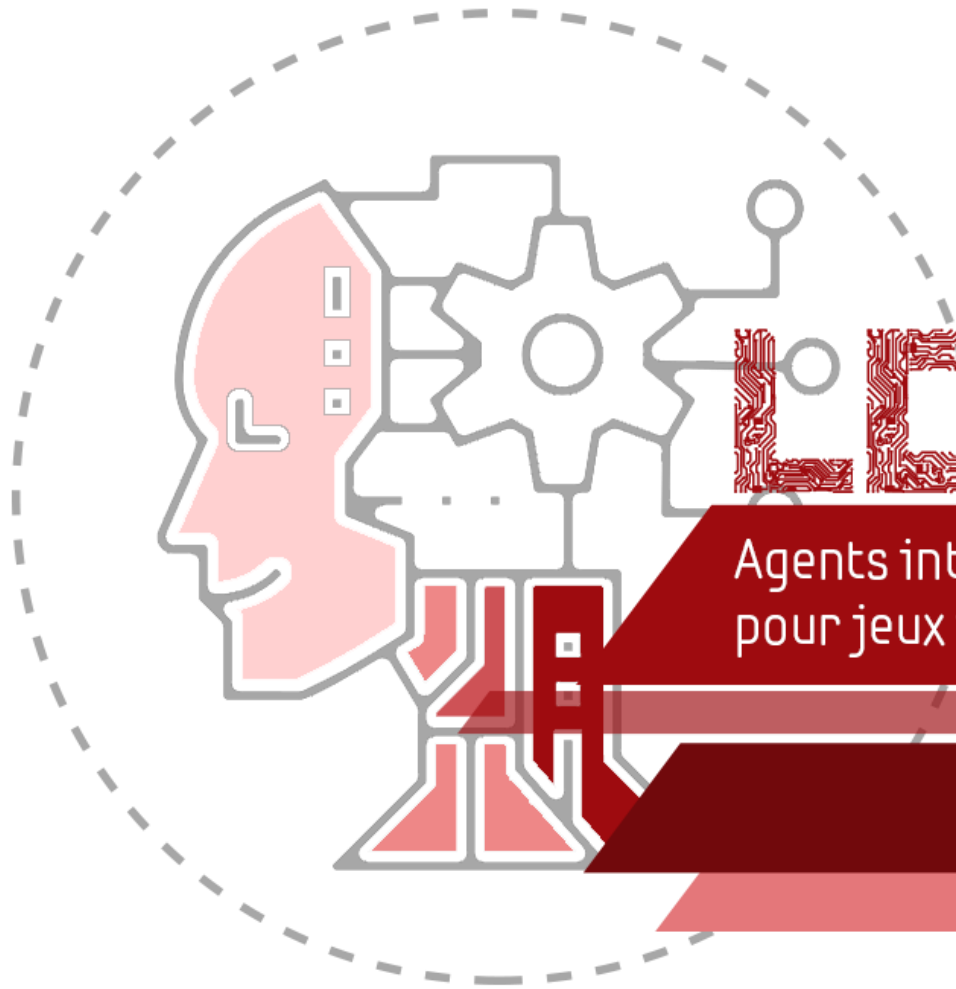
SoftDesignTraining.uproject

PS: Assurez vous que votre code est bien commenté et lisible. Cela va m'aider à vous aider lors de ma correction.

Bonne remise! 😊



POLYTECHNIQUE
MONTRÉAL



LOG 8235

Agents intelligents
pour jeux vidéo