

LOG8235

AGENTS INTELLIGENTS  
POUR JEUX VIDÉO

/ SEMAINE 1

# TRAVAUX PRATIQUE 2

# INTRODUCTION



# / INTRODUCTION

Le but de ces travaux pratiques est de mettre en pratique les différentes notions abordées lors des différents modules de cours théorique.

Les travaux doivent être réalisés à l'aide du cadriciel 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 dont les dalles de type « Death ».

Éléments obligatoires	Points
Prise en compte des dalles de type « Death » dans le navmesh généré	5 pts

# / TRAVAIL À ACCOMPLIR

## 2. Calcul et affichage d'un chemin

L'agent doit pouvoir calculer un chemin jusqu'au pickup le plus proche. Le chemin ainsi calculé devra être affiché en utilisant les fonctions de debug d'Unreal.

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

# / TRAVAIL À ACCOMPLIR

## 3. Parcours d'un chemin

L'agent doit pouvoir parcourir le chemin calculé vers le pickup le plus proche. (Aller voir le fichier USDTPathFollowingComponent.cpp)

Éléments obligatoires	Points
Parcours des différents points du chemin	5 pts



# / TRAVAIL À ACCOMPLIR

## 4. Rajout de navlink

La génération du navmesh doit prendre en compte la présence des dalles de type « Death » de façon à générer des navlinks pour que l'agent puisse sauter par-dessus.

Éléments obligatoires	Points
Rajout de navlink dans le navmesh généré	5 pts

# **/ TRAVAIL À ACCOMPLIR**

## **5. Parcours d'un chemin comportant des navlinks**

**L'agent doit pouvoir parcourir un chemin dont certains points nécessitent que l'agent saute afin d'éviter un obstacle.**

<b>Éléments obligatoires</b>	<b>Points</b>
Saut de l'agent au-dessus des dalles de type « death »	5 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. Lorsque le joueur n'est plus visible, l'agent doit se diriger vers la dernière position visible.**

<b>Éléments obligatoires</b>	<b>Points</b>
Poursuite du joueur en parcourant le chemin vers le joueur ou sa dernière position visible	5 pts

# **/ TRAVAIL À ACCOMPLIR**

## **7. Mis à jour du comportement de fuite**

**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.**

<b>Éléments obligatoires</b>	<b>Points</b>
Éloignement du joueur en parcourant le chemin vers le point de fuite le plus pertinent	5 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 supportant les vitesses walk, run	5 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.**

<b>Éléments obligatoires</b>	<b>Points</b>
Graphe d'animation supportant le saut	5 pts

# ÉLÉMENTS FOURNIS



# / ÉLÉMENTS FOURNIS 1/2

## 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).

### **SDTUtils**

- 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	5 pts
4. Rajout de navlink	5 pts
5. Parcours d'un chemin comportant des navlinks	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 realisation*	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**

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

- Taille des fonctions**
- Factorisation du code**

# REMISE



# / REMISE

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

Config/...

Content/...

Source/...

SoftDesignTraining.uproject

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

Si vous avez des éléments additionnels à mentionner, vous pouvez ajouter un README.md dans votre zip avec des instructions spéciales.

Bonne remise! 😊

LOG8235

AGENTS INTELLIGENTS  
POUR JEUX VIDÉO