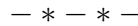


Algorithmes & Systèmes Numériques - Projet



Projet n°1 : Ultimate Tic-Tac-Toe



I - Présentation du sujet

L'*ultimate tic-tac-toe* est une extension du tic-tac-toe (figure 1) avec 9 grilles. L'objectif pour un joueur est de réussir à aligner 3 grilles gagnantes sur la super-grille. Par exemple, le joueur o a gagné sur la figure 2 en complétant les trois grilles milieu-gauche, milieu et milieu-droite.

o		x
	o	o
x	x	x

Figure 1 – Le tic-tac-toe original

	o		x	x	o		x	o
x	o			o		x	x	x
x	o	o	o	x	o	o		
o	o	o	x	o	x	x		
				o	x			
				o	o	o	o	o
x	o	o	o	x		x	x	x
x				x	x	x		o
x	o	x	x		x	o	o	x

Figure 2 – L'ultimate tic-tac-toe

Cependant, il n'est pas toujours possible de jouer sur n'importe quelle case. Si chaque case est numérotée de la manière

suivante :

1	2	3
4	5	6
7	8	9

La position choisie par le joueur précédent détermine la sous-grille sur laquelle doit jouer son adversaire : la position de la sous-grille dans la super-grille doit être la même que la position de la case dans la sous-grille au coup précédent. Par exemple, si le premier joueur commence par la proposition suivante, son adversaire devra jouer sur une des neuf cases de la grille en bas à gauche.

Si la sous-grille considérée est déjà remplie ou si un joueur y possède déjà trois cases alignées, alors le joueur courant peut jouer où il le souhaite parmi les cases libres des grilles disputées restantes.

Le jeu s'achève lorsqu'un joueur a complété trois sous-grilles alignées ou lorsque aucun coup valide n'est possible.¹

II - Travail demandé & Rendus

Ce projet s'appuie sur 3 séances encadrées (semaine du 29/11, du 03/01 et 17/01). À chaque séance, vous aurez un travail à réaliser et un rendu, en rapport avec celui-ci, à fournir à l'issue de celle-ci. Vous pouvez bien évidemment « prendre » de votre temps personnel pour terminer et/ou améliorer le travail de la séance précédente, et même commencer la réflexion sur le travail à réaliser la séance suivante !!

Objectifs du projet : À l'issue du projet, nous souhaitons disposer d'un jeu d'Ultimate Tic-Tac-Toe possédant les fonctionnalités suivantes :

- ↪ la visualisation du jeu (i.e. plateau, informations, etc.) sous forme textuelle, à la console ;
- ↪ l'enregistrement et le chargement d'une partie permettant l'arrêt et la reprise de celle-ci. Pour ce faire, les informations nécessaires seront enregistrées/chargées dans/depuis un fichier.
- ↪ *un mode jeu à 2 joueurs (humains)* : dans ce mode, vous devez permettre à chaque joueur de renseigner ses actions depuis la console. À la suite de chaque action de jeu, vous effectuerez une visualisation de l'état de la grille.
Un joueur sera modélisé par un nom, un pseudo et âge.
- ↪ *un mode jeu à 1 joueur (humains VS Machine)* : ce mode intégrera une petite et simple « intelligence artificielle ». Celle-ci jouera aléatoirement, mais en respectant les règles du jeu. Elle interagira avec le moteur du jeu à l'aide des mêmes sous-programmes que l'interface en ligne de commande.
- ↪ *un mode jeu automatique (Machine VS Machine)* : si le temps le permet, votre projet proposera un mode de jeu dans lequel deux « intelligences artificielles » s'affrontent. Vous pourrez également réfléchir et proposer des améliorations de la manière dont cette « intelligence » opère.

1. Faire une ou deux parties pour vérifier votre bonne compréhension des règles !

Planning du projet : Pour vous guider quelque peu dans ce travail, bien que l'un des intérêts de ce projet soit de vous laisser œuvrer avec une certaine autonomie², voici la déclinaison des objectifs sur chacune des séances, ainsi que les rendus à fournir.

Vous noterez que le programme de travail de ces séances est assez chargé. Travaillant en binôme, il va de soi que vous devrez, dès que possible, vous répartir le travail afin de gagner en efficacité. . .

Objectifs

Semaine du 29/11/2021

Durant cette séance, et après avoir lu l'intégralité des informations sur le moodle et de ce sujet, vous devrez :

1. Concevoir l'intégralité de la structure de données permettant de modéliser ce jeu.
2. Implémenter, en Ada, cette structure de données. Comme durant le premier TP, tous les types seront déclarés dans un package `Type_Ultimate_TicTacToe`, i.e. dans un fichier `type_ultimate_tictactoe.ads`.
3. Réaliser un programme de test de votre structure de données. À la manière du premier TP, il s'agit ici de tester l'écriture et la lecture dans/depuis une variable et ceci, pour chacun des types proposés.

À l'issue de la séance, vous déposerez sur le moodle et dans une unique archive `tictactoeS1-Nom1-Nom2.zip` :

- ↪ le répertoire de votre projet préalablement nettoyé à l'aide de la commande `gprclean` ;
- ↪ un document pdf dans lequel vous expliquerez vos choix de conception de la structure de données.

Objectifs

Semaine du 03/01/2022

Durant cette deuxième séance, dans un premier temps :

1. Si nécessaire, finaliser, au plus vite, l'implémentation de votre structure de données et les tests de celle-ci.

Puis vous vous intéresserez à la conception des grandes fonctionnalités de votre projet :

2. Pour chacune des grandes fonctionnalités de votre projet, proposer un arbre de décomposition en vous basant sur les règles de l'analyse descendante.
3. Dans le fichier `type_ultimate_tictactoe.ads`, ajouter alors toutes les signatures des sous-programmes mis en exergue durant la phase précédente.
4. Si le temps le permet, débiter la spécification et l'implémentation de l'ensemble des sous-programmes ajoutés. Les implémentations de tous les sous-programmes seront réalisées dans le fichier `type_ultimate_tictactoe.adb`.

À l'issue de la séance, vous déposerez sur le moodle et dans une unique archive `tictactoeS2-Nom1-Nom2.zip` :

- ↪ le répertoire de votre projet préalablement nettoyé à l'aide de la commande `gprclean` ;
- ↪ un document pdf dans lequel vous présenterez vos différents arbres de décomposition ainsi que la liste des signatures des sous-programmes. Pour chacun d'eux vous ajouterez un petit texte expliquant leur rôle.

Objectifs

Semaine du 17/01/2022

Durant cette dernière séance, les objectifs seront les suivants :

1. Terminer la spécification et l'implémentation des différents sous-programmes.
2. Pour chaque sous-programme implémenté, créer un petit programme de test. Les tests effectués devront être déduits des spécifications du sous-programme.
3. Réaliser le programme principal, dans un fichier `jeu_ultimate_tictactoe.adb`. Il s'appuiera bien évidemment sur les différents sous-programmes réalisés précédemment et devra répondre le mieux possible aux objectifs globaux du projet.

À l'issue de cette séance, il ne vous est demandé aucun rendu.

Vous disposez de deux semaines pour finaliser votre travail. Le rendu final demandé est décrit dans le cadre « recommandations » ci-dessous.

2. critère pris en compte dans la notation

Recommandations :

Ce projet doit vous permettre de mettre en œuvre les nombreux et importants principes vus en cours. Ainsi vous devrez porter une attention toute particulière sur les points suivants :

- ★ **la modélisation du problème** ; point incontournable pour la bonne réalisation des fonctionnalités demandées dans votre projet, vous devrez concevoir une structure de données la plus juste et abstraite possible.
- ★ **l'analyse descendante** ; pour chacune des grandes fonctionnalités mises en œuvre dans votre projet, vous devrez réaliser cette analyse afin d'en déduire les différents sous-programmes à réaliser.
- ★ **Spécifications & tests** ; Chacun de ces sous-programmes devront être spécifiés en utilisant le formalisme vu en cours. Vous réaliserez alors l'ensemble des tests unitaires qui en découlent.

À produire à la fin du projet au plus tard le 06/02/2022 - 20 h :

À l'issue de la dernière séance de projet, vous déposerez sur le moodle (à l'aide de la section de dépôt concernant votre sujet de projet) de l'école une unique archive portant le nom des deux élèves constituant le binôme, et contenant :

- ★ l'ensemble de votre code source (qui devra être commenté) ;
- ★ un rapport au format Pdf reprenant le contenu des documents déjà produits, c'est-à-dire :
 - ↪ les explications des choix de conception effectués lors de la réalisation de votre structure de données ;
 - ↪ pour chacune des grandes fonctionnalités, l'arbre de décomposition issu de l'analyse descendante faisant ressortir les différents sous-programmes réalisés ;
 - ↪ pour chaque sous-programme, la spécification, la signature et les jeux de tests effectués.

† † †