

PROJET DE SEMESTRE

L'objet du projet est de mettre en œuvre les savoirs enseignés ce semestre, et plus particulièrement :

- Gestion de Projet Informatique (M2204),
- Bases de la Programmation et Conception Objet (M2103/M2104),
- Introduction aux IHM (M2105).

De ce fait, la note finale de ce module résulte de la moyenne de trois évaluations distinctes (GP/COO/IHM).

Pour chacune des parties, il y a un enseignant référent auquel s'adresser pour les compléments d'informations en rapport :

- GP : Guillaume Baes,
- COO : Eric Leprêtre,
- IHM : Géry Casiez.

De même, même si la partie gestion de projet couvre l'intégralité du semestre, le déroulement du projet comprend trois parties successives, couvrant chacune à peu près un tiers du semestre :

1. Gestion de projet : spécifications fonctionnelles,
2. POO/COO : Conception et développement du projet en mode texte,
3. IHM : Ajout d'une interface graphique élaborée.

Ce projet est à réaliser en équipes, 6 ou 7 étudiants d'un même groupe de TD, constituées lors d'une séance de projet.

Tout au long du projet des livrables vous seront demandés, à déposer sur la plate-forme Moodle.

Pour le développement vous utiliserez le gestionnaire de versions GitLab mis à disposition par le département.

ENONCÉ DU PROJET : VIRTUAL WAR

Pour éviter les morts inutiles, l'**O.N.U.** a décidé de créer une application informatique pour remplacer les affrontements réels.

L'idée est de développer un environnement virtuel qui permette à deux pays de s'affronter, sans engager de troupes sur le terrain.

RÈGLES DU JEU

Les pays s'affrontent en faisant combattre des **robots** sur un **plateau**.

Le jeu se déroule tour par tour, une équipe est choisie aléatoirement pour commencer la partie.

Au départ les robots se trouvent dans leur base respective.

A chaque tour de jeu, chaque équipe choisit un de ses robots pour réaliser une action (*déplacement ou attaque*).

Une équipe ne peut pas passer son tour, tant qu'un de ses robots possède assez d'énergie pour réaliser une action elle doit jouer.

Au cours de la partie chaque équipe doit conserver au moins un robot hors de sa base.

La partie se termine dès qu'une des deux équipes ne possède plus de robot vivant.

LES ÉQUIPES

Le jeu oppose deux pays représentés par deux **équipes** composées d'un même nombre de robots, pour une évidente raison d'équité.

Cependant, il existe 3 catégories de robots (tireur/piégeur/char) et chaque pays choisit quels robots constituent son équipe, cela fait partie de sa stratégie pour remporter la victoire.

Les équipes sont composées au minimum d'un robot, mais ne comportent pas plus de 5 robots.

LES ROBOTS

Chaque robot appartient à une des catégories suivante :

- **tireur** : robot léger, capable de tirer à une courte distance,
- **piégeur** : robot léger, capable de poser des **mines**,
- **char** : robot lourd, capable de tirer à une distance plus grande que le tireur.

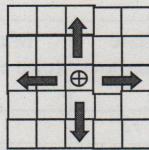
Les robots disposent d'une **énergie** initiale :

- chaque **action** leur fait perdre un peu d'énergie,
- un retour en **base** leur permet de régénérer des points d'énergie.

Les **actions** possibles sont les suivantes :

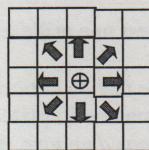
1) **Le déplacement :**

- Les **chars** se déplacent en ligne droite, horizontalement ou verticalement de **deux** cases :



- *Un char ne peut pas délibérément se déplacer que d'une seule case, mais il peut y être contraint par les limites du plateau ou un obstacle.*
- *Un char ne peut passer que par des cases vides ou des mines (auquel cas la mine disparaît et il subit des dégâts), il ne peut pas "survoler" des autres robots.*

- Les **autres robots** (tireur/piégeur) se déplacent d'une seule case dans toutes les directions.



2) **L'attaque :**

Les robots en base ne peuvent pas lancer d'attaque.

- **le tir pour les tireurs/chars**
 - Le tir est soit horizontal soit vertical et a une portée limitée.
 - Le robot qui tire perd de l'énergie, ainsi que le robot cible évidemment.
 - Le tir est arrêté par le premier élément rencontré, robot ou obstacle (il ne poursuit pas sa course).
 - Le tir n'est possible qu'en direction d'une cible identifiée.
 - Il n'est pas possible de tirer sur un robot de son équipe.
 - Le tir n'a pas d'impact sur les obstacles.
- **poser une mine** pour les piégeurs
 - La mine ne peut être posée que sur une case vide contiguë au piégeur, dans l'une des 8 directions.
 - Une seule mine peut être posée par tour.
 - Un piégeur dispose de 10 mines au départ, il doit retourner en base pour reconstituer son stock (réapprovisionnement instantané dès le retour en base).
 - Un robot qui passe sur une mine, qu'elle ait été posée par son pays ou non, perd de l'énergie (et détruit la mine).

Un robot ne peut réaliser une action que s'il possède l'énergie nécessaire.

Lorsqu'un robot n'a plus d'énergie, il disparaît du plateau de jeu.

Lorsqu'un pays n'a plus de robot vivant, il a perdu la guerre.

Au cours d'un tour de jeu, au cas où plus aucun robot n'est vivant dans les deux équipes, (*perte simultanée des derniers points d'énergie*), la dernière équipe ayant joué est celle remportant la victoire.

Le jeu est reconfigurable, la pondération des actions peut être redéfinie.

Dans un premier temps vous utiliserez les valeurs du tableau ci-dessous, associant des valeurs aux différentes actions selon les catégories de robot :

	Portée (en case)	Déplacement (en case)	Energie Initiale (en pts)	Base (en pts/ tour)	Miner (coût en pts)	Tirer (coût en pts)	Avancer (en pts)	Dégâts Tir ou Mine (en pts)
Tireur	3	1	40	+2		-2	-1	-3
Piégeur	1	1	50	+2	-2		-2	-2
Char	10	2	60	+2		-1	-5	-6

Pour chaque robot un niveau d'énergie est attribué au départ. Le seul moyen pour un robot de récupérer de l'énergie est un retour en base. Le niveau d'énergie augmente pour chaque tour passé en base mais est plafonné au niveau d'énergie initial.

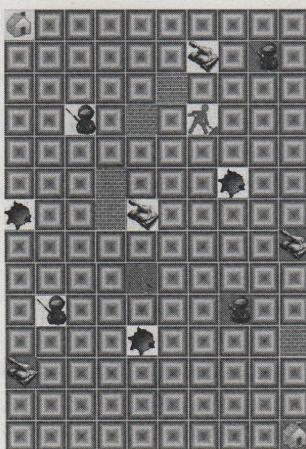
LE PLATEAU

L'affrontement se déroule sur un plateau de jeu, qui est une grille à deux dimensions dont la taille est paramétrable (*pas nécessairement carrée*).

Le nombre d'obstacles est configurable (*pourcentage fixé au départ par rapport au nombre total de cases*).

Les bases sont situées aux extrémités d'une des diagonales du plateau.

Il n'est pas possible de sortir du plateau (*il ne s'agit pas d'un tore, pas de sortie à droite pour se retrouver à gauche*).



Remarques :

- Dans leur base, les robots ne peuvent pas être touchés, et ne sont pas visibles.
- Il n'y a jamais plus d'un robot dans une case, sauf dans la base où plusieurs robots peuvent être présents.
- Un robot ne peut pas entrer dans la base adverse.
- La mine disparaît lorsqu'elle endommage un robot/char.
- Un char qui passe sur une mine au cours de son déplacement subit des dégâts.
- Il conviendra de prévoir un affichage du niveau d'énergie pour chacun des robots.
- Une équipe ne voit que ses propres mines, pas celles de son adversaire.

LES BASES

Au départ, les robots de chaque équipe sont situés dans leurs bases, situées aux deux extrémités d'une des diagonales.

C'est le seul moment du jeu où tous les robots sont dans leur base, il faut en permanence que chaque équipe ait au moins un robot hors de sa base au cours de la partie.

Lorsqu'ils sont dans leur base les robots sont invisibles et ne peuvent être attaqués (ils ne peuvent pas non plus attaquer d'ailleurs).

La base a une autre propriété intéressante au cours du jeu : lorsqu'un robot rejoint sa base son énergie se régénère jusqu'à ce qu'il retrouve son niveau d'énergie initial. Le robot récupère un peu d'énergie à chaque tour de jeu.

La base permet également aux piégeurs qui y rentrent de reconstituer leur provision de mines, stock reconstitué instantanément.

LES OBSTACLES

Des obstacles peuvent être disposés aléatoirement sur le plateau.

Un obstacle ne peut être traversé, il doit être contourné. Il arrête les tirs.

Si on choisit de placer des obstacles, le nombre de cases en comportant sera choisi par un pourcentage.

- Par exemple, si on choisit 20%, cela indique que 20% des cases, aléatoirement choisies, seront occupées par des obstacles.
- Bien sûr, il faut vérifier que le plateau obtenu peut servir pour le combat. On estime qu'un plateau est valide s'il existe un chemin entre les deux bases.

TOUR DE JEU

Un tour de jeu est composé de la séquence suivante :

- on tire au sort un pays,
- le pays tiré au sort choisit un robot qui va jouer, et une action à réaliser,
- on vérifie que le jeu n'est pas fini,
- l'autre pays choisit aussi un robot et une action,
- on vérifie que le jeu n'est pas fini.

Chaque équipe a une vision partielle du plateau de jeu : il voit si une case peut être traversée, si un robot est présent sur cette case, mais pas s'il y a une mine adverse dessus.

COMMENT JOUER

Pour ce qui est du développement on souhaite qu'il soit possible de jouer selon différents modes :

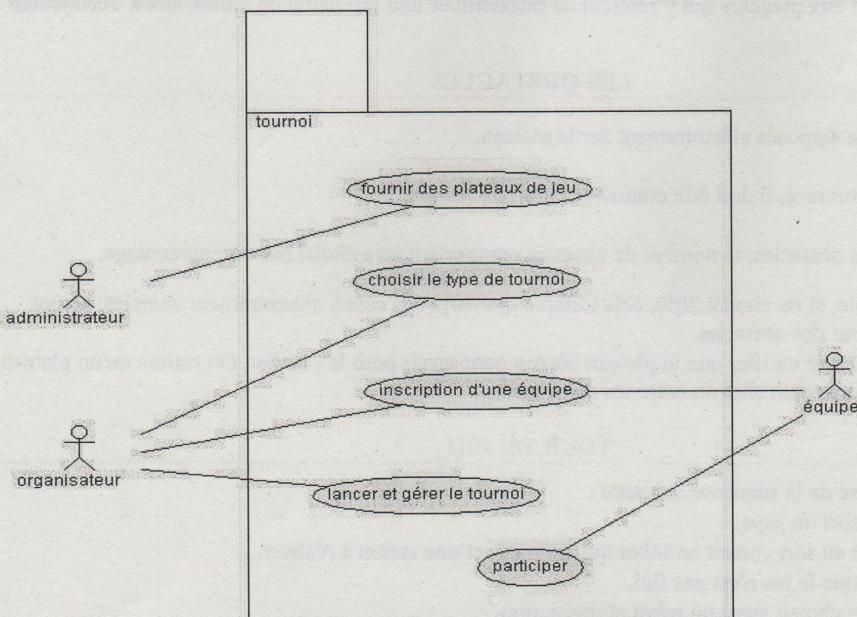
- soit de jouer manuellement contre l'ordinateur (contrôle manuel des actions des robots),
- soit de constituer une équipe et de la laisser affronter l'ordinateur,
- soit de constituer deux équipes et de les faire s'affronter.

PROTOTYPAGE

Pour aller plus loin dans le prototypage de l'application, on envisagera la possibilité d'organiser et de gérer un "tournoi" entre différentes équipes.

A cet effet, on pourra distinguer plusieurs types d'intervenants ("acteurs"):

- un administrateur dont le rôle sera de fournir à la demande des plateaux de jeu conformes à certaines spécifications, plateaux de jeu "valides" ("jouables"),
- un organisateur du tournoi chargé de choisir un type de tournoi (championnat, à éliminations, ...), recueillir les demandes de participation d'équipes, inscrire les équipes, gérer le déroulement du tournoi (définir et lancer les affrontements, gérer les résultats, ...),
- des équipes qui s'inscriront au tournoi, choisiront leurs robots, leur stratégie et affronteront d'autres équipes.



L'objectif, ici, n'est pas de développer tous les outils nécessaires à cette organisation et gestion d'un tournoi, mais consiste à spécifier, prototyper ces outils en termes de diagrammes de classes, d'interfaces, ...

CONCEPTION ORIENTÉE OBJETS

Dans cette étape (*jalon N°1 & jalon N°2*) l'affichage est à réaliser en **mode texte**.

ETAPES DE CONCEPTION

Pour faciliter l'évaluation comparative de vos projets, nous vous demandons de **respecter l'ordre** dans lesquels sont introduits les différents éléments constitutifs du jeu. Il se peut que lors de la première évaluation, vous n'avez pas eu le temps d'implémenter tous les éléments, le nombre de fonctionnalités implémentées sera un critère de notation.

De plus, il vous est demandé pour ce premier Jalon de **respecter strictement le diagramme de classes UML fourni**, ceci quelque soit ce que vous en pensez, vous aurez le loisir de le remettre en cause lors des développements ultérieurs si vous le souhaitez, en justifiant vos choix évidemment.

JALON N°1 : ÉLÉMENTS DU JEU

Cette étape donne lieu à une 1^{ère} évaluation
(*sous forme d'une démonstration en salle TP + rendu Moodle*)

Important :

Dans cette première étape :

1. l'équipe est constituée d'un robot de chaque type, au fur et à mesure de leur définition.
2. le contrôle des robots est purement manuel :
à chaque tour de jeu un menu texte permet de choisir un des robots, puis l'action à lui faire réaliser;
(les robots seront dotés d'un "cerveau" mettant en œuvre une stratégie au jalon 2).

L'objet de ce premier jalon est d'introduire un par un les différents éléments du jeu.

N°1 : La grille

Le plateau de jeu est une grille de taille paramétrable, sans obstacle, avec les deux bases (b/B) situées aux extrémités opposées d'une des diagonales.

Exemple de grille 5x10 :



Exemple de grille 10 x 15, et sa légende, intégrant tous les éléments du jeu :

B																			
	T		x	x															
					P														
	T										c								
						O	O												
						C								c					
																		t	
																			O
																			o
																			o
																			o
																			b

MAJUSCULES	%	minuscules	
B		b	: base
T		t	: tireur
P		p	: piègeur
C		c	: char
X		x	: mine

O		:	obstacles

L'affichage sera complété par :

1. une représentation de l'énergie de chacun des robots,
2. le nombre de mines en réserve pour les piègeurs,
3. un message informatif sur le résultat de l'action (succès, échec...)

Les robots démarrent leur partie depuis leur base respective.

Pas de retour en base possible dans un premier temps (*et donc pas de régénération de l'énergie*).

N°2 : Le robot Tireur

Les deux équipes sont constituées d'un seul robot tireur (t/T).

Le déplacement des robots (*une case, 8 directions*) et le tir (*horizontal ou vertical, portée limitée*) sont contrôlés grâce à un menu en mode texte.

N°3 : Le robot Piégeur et la mine

Les deux équipes sont constituées d'un tireur (t/T) et d'un piégeur (p/P) pouvant poser des mines (x/X).

N°4 : Le char

Les deux équipes sont constituées d'un tireur (t/T), d'un piégeur (p/P) et d'un char (c/C).

Le déplacement (*horizontal ou vertical, portée deux cases*) des robots est contrôlé grâce à un menu (en mode texte).

N°5 : Les obstacles

Le plateau comporte des obstacles (*en nombre paramétrable*) disposés aléatoirement lors de sa création.

On prendra soin de vérifier que le plateau est valide : qu'il existe au moins un chemin reliant les deux bases.

N°6 : Régénération de l'énergie/Retour en base

Les robots peuvent retourner dans leur base (*où ils sont invulnérables*), pour régénérer leur énergie ou/et reconstituer leur stock de mines pour les piégeurs.

JALON N° 2 : JEU COMPLET EN MODE TEXTE

Cette étape donne lieu à une 2^{ème} évaluation COO
(sous forme d'une démonstration en salle TP + rendu Moodle)

A partir de maintenant vous pouvez si vous le souhaitez adopter une autre modélisation UML que celle préconisée au jalon 1.

Les équipes représentant les pays sont constituées de 5 robots pouvant être des tireurs, des chars ou des piégeurs.

- Vous êtes libres quant à la façon de désigner les pays qui vont s'affronter,
- Chaque pays configure son équipe en choisissant ses 5 robots parmi tireurs chars, piégeurs (*prévoir une désignation aléatoire pour pouvoir affronter l'ordinateur*).

A chaque tour de jeu un robot de chaque équipe réalise une action. Dans cette phase le choix du robot est soit manuel, soit purement aléatoire, on ne demande pas de mettre au point de stratégie d'équipe pour la désignation du robot qui va réaliser l'action.

Les robots sont soit contrôlés manuellement, soit dotés d'un cerveau pour décider eux-mêmes de l'action à mener lorsque c'est leur tour de jeu.

Les robots peuvent donc mettre en œuvre une stratégie individuelle. Chaque robot choisit selon des critères qui lui sont propres et la vue qu'il a du plateau de jeu, une des actions parmi celles qu'il peut légitimement réaliser à cet instant.

Lors de leur création les robots sont configurés avec un cerveau particulier, chaque "cerveau" correspond à une stratégie de choix d'action.

A vous de définir des stratégies plus ou moins sophistiquées.

La stratégie la plus triviale réside en choix purement aléatoire parmi toutes les actions possibles.

Quelques pistes pour des stratégies plus élaborées :

- offensive : privilégiant l'attaque,
- défensive : privilégiant la fuite,
- économique : préservant les niveaux d'énergie,
- robot muni d'une mémoire de sa (ses) dernière(s) action(s) réalisée(s)...

EVALUATION COO FINALE :

- Du point de vue Conception Orientée Objet les points évalués, sont principalement :
 - le diagramme UML,
 - la structuration du code,
 - fichiers properties renfermant les paramètres du jeu,
 - la javadoc fournie,
 - les tests mis en œuvre,
 - la gestion des exceptions,
 - ...
- Seront pris en compte également :
 - la sophistication des stratégies,
 - les développements complémentaires.

AUTRES DÉVELOPPEMENTS POSSIBLES

Ci-après nous vous proposons quelques pistes de développement pour compléter votre projet.
Vous pouvez approfondir l'un ou l'autre de ces thèmes, ou une combinaison partielle de ces derniers.
Vous pouvez également proposer vos propres améliorations, par exemple :

- case téléportation,
- bonus vie ou armement dans certaines situations,
- prise en compte de l'expérience pour faire évoluer l'armement,
- possibilité d'ajouter de nouveaux types de robots, configurer le plateau...

JEU AVEC STRATEGIE D'EQUIPE

Jusqu'à présent, les équipes, autres que celles en mode manuel, n'ont, en dehors du choix initial des robots, aucun contrôle sur le déroulement d'une partie.

L'objectif est maintenant de permettre aux équipes de contrôler le choix du robot devant agir lors d'un tour de jeu, plutôt que de laisser ce choix au seul hasard. Les robots conservent toutefois leur propre "cerveau" mis à contribution par l'équipe.

Différentes solutions sont envisageables comme par exemple :

- l'équipe choisit (selon ses propres critères) le robot devant agir, ce robot choisissant alors seul l'action qu'il réalisera (contrôle centralisé, puis choix individuel);
- chaque robot propose à l'équipe l'action qu'il souhaite réaliser, l'équipe choisissant alors l'une de ses propositions (choix individuel, puis contrôle centralisé);
- l'équipe choisit le robot et l'action à réaliser par ce robot (contrôle absolu de l'équipe, les robots n'ont plus d'initiative)...

La suite du projet consiste à développer une interface graphique en mettant en œuvre les préceptes vus en cours.