1.2 Les Classes.

Dans un premier temps, nous avons choisi de faire la distinction entre les cartes, les robots et les joueurs, en les définissants dans des classes différentes plus simple à créer. L’objectif étant de créer des fonctions pour les faires interagir entre elle lors du déroulement du jeu.

Nous avons donc défini trois classes principales : les cartes, les robots et les joueurs.

1.2.1 La classe « carte » :

Il existe dans ce jeu quatre types de cartes : « avancer », « reculer », « tourner » et « demi-tour ».

On peut avancer de une, deux ou trois cases. On peut reculer. On peut tourner d’un quart de tour vers la gauche ou vers la droite. On peut faire demi-tour.

Nous avons donc défini la classe « carte » avec quatre classes héritières.

Le constructeur de la classe « carte » est défini par quatre items :

* La priorité ;
* Le nombre de rotation ;
* Le nombre de case ;
* Le numéro du joueur à laquelle elle appartient.

Chacun des types de cartes est donc caractérisé par ces quatre items.

Les cartes de types « avancer » et « reculer » ont un nombre de rotation égale à zéro, les cartes « avancer » ont un nombre de case variant de 1 à 3 et la carte « reculer » a un nombre de case égale à -1.

Les cartes « tourner » ont un nombre de case égale à zéro, un nombre de rotation égale à 1 indique une rotation vers la droite et un nombre de rotation égale à -1 une rotation vers la gauche.

La carte « demi-tour » a un nombre de case nul et un nombre de rotation de 2.

1.2.2 La classe « robot » :

Chaque joueur dispose d’un robot qu’il déplace à l’aide de ses cartes vers la ligne d’arrivée. Nous avons choisi de faire la distinction entre un joueur et son robot pour simplifier le programme et éviter de créer une classe trop complexe.

Le constructeur de la classe « robot » est défini par quatre items :

* La vie ;
* La direction de départ ;
* Les coordonnées ;
* Le plateau : nous avons choisi de lier le robot au plateau afin de faciliter les mouvements de celui-ci au cours de la partie.

Le robot peut avancer, reculer et tourner. Nous avons décidé de créer des fonctions, au sein de la classe « robot », qui ont pour objectif de modifier les caractéristiques du robot en fonction de la carte jouée.

On a alors une fonction « Turn », qui modifie la direction du robot en prenant en entrer le nombre de rotations de la carte jouée. On a également une fonction « Forward », qui en prend en compte la direction du robot et fait évoluer les coordonnées du robot en fonction du nombre de case de la carte jouée.

1.2.3 La classe « joueur » :

Chaque joueur a autant de carte dans sa main que de point de vie et est lié à son robot. Au début de la partie le joueur choisit un robot qui est placé aléatoirement sur le plateau de jeu.

Pour s’assurer que le vainqueur ait bien respecté les conditions de victoire en passant dans l’ordre par les différents check point nous avons créé dans le constructeur de la classe « joueur » un item « fil rouge » qui prend en compte cet aspect du jeu.

Le constructeur de la classe « joueur » est donc défini par quatre items :

* Le nombre dans la main du joueur ;
* Le nombre de cartes jouées ;
* Le robot ;
* Le fil rouge, qui prend la valeur 1 si les conditions de victoire sont vérifiées et déclenche la fin de la partie.

La classe « joueur » est également composée de deux sous classes : Utilisateur et IA, que nous n’avons pas encore définies.

1.3.1 La phase d’initialisation :

La fonction initialisation remplit ce rôle en prenant en entrée le nombre d’IA et le nombre de joueur. Elle attribue à chaque joueur et à chaque IA son numéro de joueur et crée deux listes « Bot » et « Joueur » où elle les range. Elle charge également le plateau.

1.3.2 Les tours :

La phase de distribution est décrite par une fonction « Distribution ». Cette fonction attribue aléatoirement à chaque joueur autant de carte que son nombre de point de vie et crée la relation d’appartenance d’une carte à son joueur. Les cartes distribuées sont rangées dans des listes qui représentent les mains de chacun des joueurs.

La phase de choix est une phase d’interaction où l’utilisateur va entrer une liste ordonnée des 5 cartes qu’il veut jouer lors de ce tour. Nous n’avons pas encore défini la fonction liée à l’IA qui sera responsable du choix des cartes.

La phase d’application est composée de deux fonctions : une fonction de tri, qui trie une liste par ordre de priorité de chaque carte. Ensuite la fonction « Application des cartes prend la première carte de chaque joueur pour les mettre dans une liste, qui est ensuite triée par la fonction de tri. La liste triée est parcourue et toute les cartes sont appliquées une à une. Cette séquence est réalisée dans une boucle qui est réalisée 5 fois.