

Projet: Colt Express

Ce projet vous invite à construire une version électronique et un peu simplifiée du jeu Colt Express. Il faut réaliser le projet en binôme et le présenter la semaine du 12 Décembre, à votre horaire habituel de TP.

1. Aperçu des règles du jeu:

Le jeu se déroule à bord d'un train, composé d'une locomotive et d'un certain nombre de wagons égal au nombre de joueurs. Les joueurs incarnent des bandits qui ont sauté à bord pour détrousser les passagers. Objectif: récupérer le plus de butin possible, chacun pour soi. Il s'agit d'un jeu de programmation, dans lequel on alterne entre deux phases :

- **Planification** : chaque joueur décide secrètement un certain nombre d'actions, que son personnage va effectuer dans l'ordre.
- **Action** : on effectue toutes les actions numéro 1, puis toutes les numéro 2, et ce jusqu'au bout.

Les bandits peuvent se trouver dans les wagons ou la locomotive, et pour chacun de ces éléments soit à l'intérieur soit sur le toit. Dans cet énoncé, par abus de langage on désignera par wagon un élément quelconque du train, qui peut être la locomotive. Les actions possibles pour les bandits sont :

- Se déplacer d'un wagon en avant ou en arrière, en restant au même étage.
- Aller à l'intérieur ou grimper sur le toit de leur wagon actuel.
- Braquer un voyageur pour récupérer du butin (ou simplement récupérer un butin qui a été abandonné là).
- Tirer sur un autre bandit proche pour lui faire lâcher son butin.

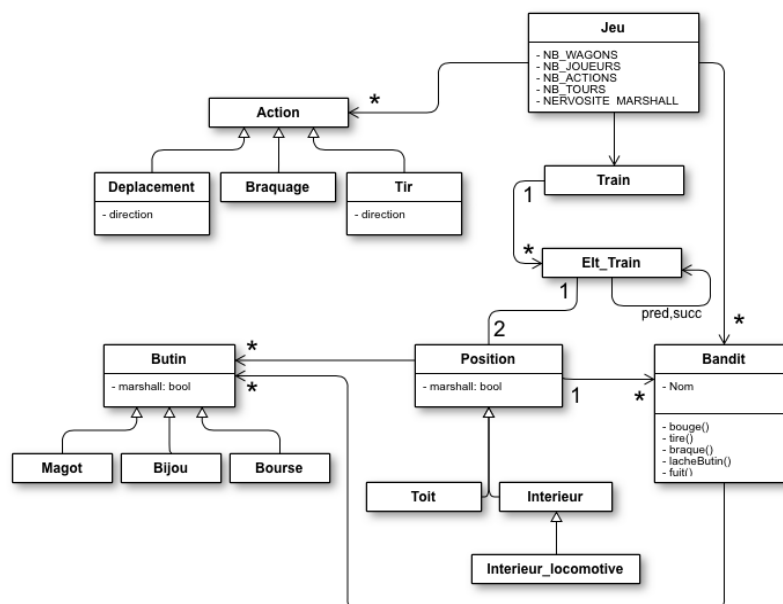
Les butins récupérables à bord du train sont :

- Des bourses valant 100€ ou 200€, auprès des passagers, à l'intérieur des wagons.
- Des bijoux valant 500€, auprès des passagers, à l'intérieur des wagons.
- Un magot valant 1000€, à l'intérieur de la locomotive, sous la garde du Marshall.

Un Marshall est présent à bord du train et peut se déplacer entre la locomotive et les wagons, en restant toujours à l'intérieur. Il tire sur tous les bandits qui se trouvent à la même position que lui et les force à se retrancher sur le toit.

2. Un modèle réduit:

Un élément central du jeu est le train lui-même, qui sert de plateau de jeu. Déterminez les différents éléments qui le composent (wagon, locomotive, ...) et écrivez les classes correspondantes. Ajoutez une classe pour les bandits et une pour les actions de déplacement. Vous pouvez vous inspirer du diagramme de classes ci-dessous (il n'est pas obligatoire de le respecter):



Note : pour l'instant, on ne s'intéresse pas aux autres actions (braquage, tir), ni aux butins ni au Marshall.

Quelques précisions :

- Le nombre des wagons est donné par une constante NB_WAGONS que vous devez définir comme attribut de la classe principale. Vous pourrez par exemple la fixer à 4.
- On placera pour l'instant un unique bandit à bord du train, sur le toit du dernier wagon. Le nom de ce bandit est donné par une constante NOM_BANDIT_1.
- Un déplacement vers l'arrière lorsque l'on est dans le dernier wagon ou en avant lorsque l'on est dans la locomotive n'a pas d'effet. De même pour un déplacement vers le toit ou vers l'intérieur lorsque l'on y est déjà.
- A chaque action du bandit, affichez un compte rendu sur la sortie standard.

Par exemple :

Joueur1 grimpe sur le toit.

Joueur1 est déjà sur le dernier wagon.

3. Une belle vue:

La partie la plus importante du projet est de réaliser une belle interface graphique.

Pour commencer, ajoutez une interface comportant deux parties :

- Un panneau d'affichage dans lequel on voit les différents éléments du train et la position du bandit.
- Un bouton "Action" qui effectue la prochaine action du bandit.

Quelques précisions :

- Pour l'instant, le bandit n'effectue qu'une suite d'actions prédéterminées.
- L'affichage doit être mis à jour à chaque déplacement du bandit.
- Continuez à afficher le compte-rendu sur la sortie standard.

4. Les récompenses:

Ajoutez dans le modèle les butins et le Marshall.

Quelques précisions :

- A l'intérieur de chaque wagon se trouvent initialement entre 1 et 4 butins (de type bourse ou bijou). La nature des butins présents à chaque position doit apparaître sur l'interface, mais pas leur valeur.
- Le magot et le Marshall sont placés à l'intérieur de la locomotive.
- Avant chaque action du bandit, le Marshall se déplace dans une direction aléatoire (mais toujours à l'étage inférieur, c'est-à-dire à l'intérieur des wagons).
- Le bandit a maintenant accès à une nouvelle action : *braquage*. Cette action lui fait récupérer un butin au hasard parmi ceux présents sur sa position.
- Si le Marshall atteint ainsi la position du bandit, le bandit lâche un des butins qu'il a ramassé tiré au hasard (s'il en a) et se déplace immédiatement vers le toit. Le butin est ajouté à l'ensemble des butins de la position dont le bandit vient d'être chassé, et peut être récupéré à nouveau.
- Continuez à afficher sur la sortie standard un compte-rendu de tous ces événements.
- Une fois toutes les actions effectuées, affichez le montant total des butins possédés par le bandit.

5. Planification des actions:

Toujours en considérant un seul bandit, ajoutez maintenant de nouveaux boutons permettant au joueur de planifier les actions de son personnage. A l'état de planification, le joueur doit donner un certain nombre d'ordres en cliquant sur les boutons correspondants. Le bouton "Action" est inopérant pendant cette phase. A l'état d'action, dans lequel chaque clic sur le bouton "Action" effectue la prochaine action, les autres boutons doivent être inopérants.

Quelques précisions :

- Le panneau de commande doit afficher une ligne de texte indiquant quelle est la phase en cours (voir par exemple la classe Label).

- Le nombre d'actions que doit indiquer le joueur lors d'une phase de planification est une constante NB_ACTIONS que vous pouvez fixer par exemple à 4.

6. Plusieurs joueurs:

Etendez maintenant le jeu pour pouvoir y jouer à plusieurs. Une vue supplémentaire devra montrer l'état de chaque bandit, en particulier les butins qu'il porte et le nombre de balles qu'il a encore à sa disposition. C'est le moment également d'ajouter aux bandits l'action de tir.

Quelques précisions :

- Le nombre de joueurs est donné par une constante NB_JOUEURS qui est égale au nombre de wagons.
- Les joueurs jouent toujours dans le même ordre.
- Lors de la phase de planification, chaque joueur à son tour donne tous ses ordres à la fois. Lors de la phase d'action, on effectue d'abord toutes les premières actions (dans l'ordre des joueurs), puis toutes les deuxièmes actions (dans le même ordre), et ainsi de suite.
- Un tir est fait dans l'une des quatre directions: en avant ou en arrière au même étage, ou vers le haut ou vers le bas dans le même wagon. Un tir vers le haut (resp. vers la bas) lorsqu'un bandit se trouve sur le toit (resp. à l'intérieur) vise la position occupée par le bandit.
- Le tir touche un bandit tiré au hasard parmi les occupants de la position visée, excepté le tireur lui-même.
- Un bandit touché par un tir lâche l'un de ses butins tiré au hasard. Le butin est ajouté à ceux présents à la position du bandit touché et pourra être récupéré à nouveau.
- Chaque tir utilise une balle. Chaque bandit possède à l'origine un nombre de balles donné par une constante NB_BALLES, que vous pourrez par exemple fixer à 6.
- Le compte-rendu textuel doit également mentionner ces actions de tir et leurs conséquences.

7. Expression libre:

Ajoutez pour finir au moins une fonctionnalité supplémentaire à votre application. Voici une mini-liste de suggestions d'ajouts. La liste n'est évidemment pas limitative et a simplement pour rôle de vous donner un point de départ:

- Récupérer tous les paramètres du jeu dans une fenêtre de dialogue s'ouvrant au début de la partie plutôt que d'utiliser des valeurs fixes.
- Lors de la planification, afficher la séquence d'ordres donnée par le joueur courant.
- Introduire des raccourcis clavier pour les différentes commandes du jeu.
- Afficher le compte-rendu de la partie non sur la sortie standard mais dans une zone dédiée de la fenêtre graphique.
- Ajouter du son et de la musique dans le jeu.
- Ajouter un bouton permettant de sauvegarder une partie, et un autre pour charger la sauvegarde la plus récente.

Consignes:

Le README détaillé doit contenir :

- Les parties du sujet que vous avez traitées.
- Les problèmes que vous avez rencontrés et que vous avez réussi à éliminer.
- Les problèmes qui sont présents et que vous n'avez pas pu éliminer.
- Les morceaux de code écrits à plusieurs ou empruntés ailleurs.
- Nous allons tester votre application : le code source doit impérativement compiler sans erreur, et au minimum afficher une fenêtre à l'exécution (même vide). Si des morceaux de code ne sont pas finis, vous pouvez les mettre en commentaire. Une méthode non implémentée peut avoir un code vide, ou renvoyer une valeur arbitraire.

Ces deux consignes (format et contenu du README, code source qui compile) sont impératives; leur non-respect équivaut à un projet non rendu et aucun droit au rattrapage. L'évaluation sera basée sur votre présentation, votre README, le test de votre application, et la lecture de passages de votre code choisis par votre encadrant.