

IPO - Rapport de projet

Trévis Morvany

Travail effectué

Les 4 premières parties ont été entièrement réalisées (Parties 1, 2, 3 + les 3 sujets de la partie 4), concernant la partie 5 j'ai légèrement modifié l'interface graphique.

Pour les sujets de la partie 4 :

- **Timer** : j'ai utilisé `System.nanoTime()` pour connaître l'instant de début et de fin de partie, pour récupérer le temps de jeux il suffit de faire : temps de jeu = tfin de partie – tdébut de partie
- **Cases spéciales** : j'ai utilisé un enum pour les types de cases et une classe `SpecialCase` qui permet de représenter une case spéciale appartenant à une Lane. Pour l'interaction de la grenouille avec ces cases j'ai modifié la méthode `isSafe` pour qu'elle renvoie un `BitSet` où chaque bit correspond à un état possible de la case, cette information est ensuite gérée dans la méthode `testLose` de `Game`.
- **Lignes d'eau** : pour les lignes d'eau j'ai dû apporter des modifications importantes à mon projet, notamment :
 - Je suis passé d'une unique classe `Lane` à 3 classes :
 - * **Lane** : permet de créer une voie vide avec si besoin une couleur particulière
 - * **CarLane** : permet de créer une voie classique avec des voitures (équivalent de l'ancienne classe `Lane`), hérite de **Lane**
 - * **WaterLane** : permet de créer une voie bleue avec des rondins à la place des voitures, hérite de **Lane**
 - Pour simplifier les choses j'ai créé une classe abstraite `Obstacle` dont héritent les classes `Car` (pour les voitures) et `Log` (pour les rondin).
 - Il suffit ensuite de déclarer une `ArrayList<Obstacle>` dans `Lane` et d'y ajouter des `Car` si on est dans `CarLane` et des `Log` si on est dans `WaterLane`.

Problèmes rencontrés

Un des points de difficulté pour la partie 2 a été l'équilibrage de la densité des voies, je n'arrivais pas à trouver une bonne valeur de densité, soit j'avais trop de voitures et c'était impossible de se déplacer, soit mes voies étaient vides, donc j'ai fini par déroger à la règle de "A chaque tic d'horloge, une voiture peut être ajoutée" ce qui m'a permis d'avoir un nombre correct de voitures. Toutefois j'ai corrigé ça une fois arrivé à la partie 5 et j'en ai profité pour améliorer la densité d'obstacles.

J'avais aussi quelques inquiétudes dans la partie 3 au niveau de la consommation de mémoire de la version infinie, j'ai donc commencé à faire une version infinie qui gardait un nombre de voies limité en mémoire (et donc supprimait les anciennes voies pour en créer de nouvelles), mais je me suis rendu compte que ça aurait été compliqué d'ajouter certaines fonctionnalités par la suite (et aussi que la consommation de mémoire n'était pas si élevée que ça), donc j'ai juste gardé une version où des lignes sont ajoutées à mesure que la grenouille avance.

J'ai également dû passer beaucoup de temps à lire la documentation et à tester des choses pour comprendre comment fonctionne l'interface graphique et pouvoir la modifier dans la partie 5.

Enfin j'ai indiqué dans le README de mon projet les bugs que j'ai rencontré (et qui ont été corrigés).