***Rapport de projet Frogger***

**Parties traitées :**

* Partie 1 par Zaineb Mian :

getPosition() : donne la position actuelle de la grenouille

getDirection() : donne la direction actuelle de la grenouille

move() : selon une direction propre à la classe Frog, est redessinée une case plus loin dans la direction voulue.

* Partie 2 par Zaineb Mian & Amale Ezzohri

*Car* possède plusieurs méthodes qui permettent de faire avancer la voiture et de savoir si une voiture ne se trouve pas hors de la grille du jeu [appearsInBounds()] ou une voiture à la case demandée, tout en actualisant dans le jeu.

*Lane* possède la méthode update() qui déplace les voitures à chaque tic de l’horloge et en même temps enlève les voitures qui dépassent le l’écran

*Environment* possède des méthodes permettant de savoir si la case est une case d’arrivée ou encore de savoir si une case est sans danger.

* Partie 3 par Zaineb Mian & Amale Ezzohri

*FrogInf*hérite de la classe Frog sauf qu’il n’y a pas de ligne d’arrivée attribuée car le jeu est infini. La méthode move() a permis de modifier cela.

**Difficultés trouvées & solutions envisagées :**

On a eu énormément de difficultés : premièrement nous n’avons jamais réussi à compiler. Deuxièmement, dans la partie 2 nous avons eu du mal à implémenter le nouvel environnement dans la Classe Main car il ne le reconnaissait pas. Dans Environment.java il a fallu en sorte de ne rien oublier. Dans la partie 3 on a rajouté par nous-même la classe FrogInf, qui ressemble beaucoup à la classe Frog. On a initialisé la laneInf dans le Environment car on ne savait pas ou le mettre exactement. Elle permet de déplacer l’environnement vers le bas ou le haut. Nous n’avons pas eu le temps d’implémenter le timer malheureusement.