

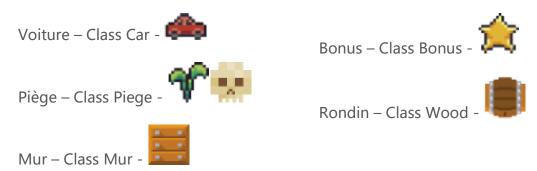
Mini Projet L2-IPO

Shuangrui CHEN | Yongjie LIU

I – Introduction:

Nous avons fini les parties 1, 2, 3 et la partie 4 entière et nous avons amélioré le rendu graphique. De plus, nous avons ajouté la musique d'ambiance, la musique de mort et l'écran d'accueil. Un nouveau class d'énumération qui s'appelle Logo pour indiquer les images à afficher. Et des nouveaux class de l'objet comme les pièges sont ajouté. Les rivières s'apparaissent 10 lignes chaque fois.

Voici les propriétés des objets : (Stocké dans ./materials)



II - Difficultés:

 Toujours des contenus de elementsToDisplay sont des pointeurs NULL, et cela se passe dans la version initiale de FroggerGraphic, mais cela n'influence pas de l'affichages, et à la fin on le jette.

```
"C:\Program Files\Java\jdk-13.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 26 Exception in thread "AWT-EventQueue-0" java.lang.NullPointerException at graphicalElements.FroggerGraphic.paintComponent(FroggerGraphic.java:156) at java.desktop/javax.swing.JComponent.paint(JComponent.java:1074) at java.desktop/javax.swing.JComponent.paintToOffscreen(JComponent.java:5255) at java.desktop/javax.swing.RepaintManager$PaintManager.paintDoubleBufferedImpl(RepaintManager.java:161) at java.desktop/javax.swing.RepaintManager$PaintManager.paintDoubleBuffered(RepaintManager.java:161)
```

- 2. Nous voulons de créer un class de père pour les class dans le package environnent, parce qu'ils ont beaucoup de méthodes similaires mais, Même s'ils sont similaires (il faut encore surcharger des méthodes), un class de père n'a pas économisé des codes donc on a restauré la version initiale.
- 3. Pour afficher le temps, on a trouvé un class Timer, mais pour afficher le String de temp, on a écrit des codes très compliqué, mais

on n'a pas trouvé une méthode pour simplifier.

```
if (second < 10 && hour >= 10 && minute >= 10)
    time = hour + ":" + minute + ":0" + second:
else if (second >= 10 && hour < 10 && minute >= 10)
    time = "0" + hour + ":" + minute + ":" + second;
else if (second >= 10 && hour >= 10 && minute < 10)
    time = +hour + ":0" + minute + ":" + second:
else if (second < 10 && hour < 10 && minute < 10)
    time = "0" + hour + ":0" + minute + ":0" + second;
else if (second < 10 && hour < 10)
    time = "0" + hour + ":" + minute + ":0" + second;
else if (second >= 10 && hour < 10)
    time = "0" + hour + ":0" + minute + ":" + second;
else if (second < 10)
    time = hour + ":0" + minute + ":0" + second;
else
    time = hour + ":" + minute + ":" + second;
```

4. Il y a beaucoup de détails qu'on doit faire l'attention, par exemple, au début, on ne réfléchit pas que la place de voiture lorsqu'on veut ajouter une nouvelle voiture, si Fristcase a déjà une voiture, on ne peut pas ajouter une nouvelle voiture, donc on écrit une fonction getFristcase() et getbeforeFirstcase() dans la fonction isSafe()pour vérifier si il y a les voitures.

III - Conclusion

Nous avons bien compris les concepts de l'Interface, des variables privées et de l'hérédité. Dans notre coopération, nous avons spontanément ajouté des méthodes pour notre binôme de récupérer des variables privées et de signifier les méthodes à implémenter dans l'interface. Nous pensons que cela c'est l'essence de ce langage, c'est de coopérer pour un projet complexe et de ne pas réinventer la roue.