Group 7 Phase 4 Report

Game Description:

Our game “Rabbit Run” is a single-player 2D arcade-style game. The objective for the main player is to navigate through the jungle-themed map and reach the exit while collecting all the regular clover rewards on the map and reaching the level's score goal. The game ends once the enemy catches up with the player or the player collects enough punishments to obtain a score of zero or less. The player also has the chance to gain bonus points by collecting the carrots that randomly appear during the gameplay.

Faithfulness to Original Plan:

We stayed quite faithful to our original design throughout the development of our game. The core elements, including the main character, enemies, and rewards, remained consistent with our initial concept.

Changes Made to Original Plan:

- Multiple Levels to a single level
  - We had initially decided that our game would have 3-5 levels that the user must finish before completing the game. With the game map becoming larger upon each subsequent level. Also the amount of enemies, punishments increases. The amount of points required to win also increases. Due to time constraints, we instead chose to create a single level for our game. This change was justified as we created a single large sized map that allowed the player to explore the playable area.

- Punishment
  - We initially planned for our punishment to be a squirrel object that would pop up randomly on the board. However, we changed the design to a poisonous mushroom because it was easier to find a suitable stock photo, allowing us to focus on other aspects of the game's development.

- Enemy Speed
  - We had initially thought to vary enemy speed based on specific conditions that the player had caused or encountered throughout the game. For example, when a player collects a punishment object, their score would be decreased and the speed of the moving enemies would increase. However when implementing this behaviour we noticed that enemies would either become too fast making the game unplayable. Also if enemy speed was too low to begin with then there would be a lack of challenge with the game. In the end we opted to leave enemy movement speed fixed to maintain playability of our game.

Lessons Learned:

- Phase 2 Initial meetings and steps
    - In our first meeting at the start of phase 2 we had tried to break down the project into parts that we could code individually. However, this proved difficult as none of our group members had experience building a project of this scale and we felt lost in the planning part of phase 2. Ultimately we were able to decide on the frameworks Java Swing and Java AWT to build our project. We also decided that in our first week of phase 2 we would individually learn Java Swing and Java AWT to familiarize ourselves with the frameworks before breaking the project down into parts that we could work on individually. Diving straight into coding our project from the beginning of phase 2 would have been an impossible task. We learned that we had to update our individual knowledge about the frameworks we had chosen to use, before we could work together as a cohesive unit. This resulted in substantial improvements in our communication as we were more knowledgeable about how the structure of our project should be built and were able to break down the project into parts that we could individually code.

- Breaking down classes and refactoring
    - In our testing phases we had problems with branch coverage being extremely low because we were only testing certain methods within classes that were too long to begin with. We learned that when classes contain too many methods they become difficult to test in isolation. Therefore in our refactoring, many classes were broken down into their key components and separated into their own classes. This improved our branch coverage results and fixed our unit testing.

Game Tutorial:

Our game demo is available through the video tutorial linked below-
https://youtu.be/_ElJut3ddKc