

## T10-G02 Final Project Description

**Category:** Animated Game

**Project:** "Simple PacMan"

**User:** moves a sprite using WASD inputs, similar to arrow keys. In both the GUI (JavaFX Application) and Text-Based (displayed in console) versions, the ghost and PacMan move with continuous, fluid movement. The user is able to give WASD key inputs in order to change the direction of the PacMan sprite's movement.

If the user's sprite collides with an enemy, the game will end. The user only has a single life. The user will also have a score counter, which will increase every time a coin is collected. The maximum number of coins to collect is 606. If the user manages to collect all of these, the game is ended and the player has won.

**Enemies:** there is only a single enemy that will chase PacMan. The reason only one enemy exists is because unlike in the original, the ghost here does not stop following the player at any time. Only having one single enemy is balanced out by the user only having a single life. This ensures the game is not too difficult and not too easy.

The ghost moves at the same speed as the player. It cannot collect coins, cannot go through walls, and follows the player by calculating the shortest distance. If the ghost catches PacMan, the game ends and the player has lost.

**Coins (objectives):** each empty tile (excluding those in the center box where PacMan is spawned) that does not contain a wall will contain a coin. The user can 'eat' these coins by simply colliding with them. If all coins are collected, the game ends and the player wins.

### **Implementation:**

- Keep track of moving and stationary items, storing each in a single list.
- The application will detect and respond appropriately to collisions between items.
- For Demo 1 we created a text-based version that mimics the animation, using and printing a 2D array of game items (walls, coins, etc). This makes it easier to keep track of correct collision detection, as well as the if the ghosts are correctly chasing PacMan. For the initial text-based version of the game, the user moved one space at a time. This means that the user was asked for movement input every time after a single move.
- For Demo 2 we began creating the GUI version that uses JavaFX and AnimationTimers in order to generate the animations. It also checks for user events (such as key presses), rather than pausing the program constantly to wait for user input. This makes for smoother gameplay.

### **Game features:**

- The player has a single life. The game ends when it is gone, or when the user has collected all coins. There is only one single level.
- One ghost will chase the player. It moves at the same speed as the player.

- The player's score increases when coins are collected. This score will be displayed at the top of the window as the game is running.
- A menu screen gives the player the option of loading a previously saved game, or starting a fresh one
- By selecting 'L' on the main menu, the user can load their last-saved game. Score, player position, and enemy positions are all stored. Coins that have not been collected appear in the same spots as when the game was saved.
- The player moves using the "WASD" keys
- Hitting the spacebar during game play will pause the game and bring up a "paused game" menu
- While paused, the player can choose to save their current game progress, resume their game, or return to the main menu where they can start a new game or load a previous game
- The game ends when either all coins have been collected, or when the enemy has caught the player
- When the game ends, the player has the option to start a new game without closing and restarting the application.