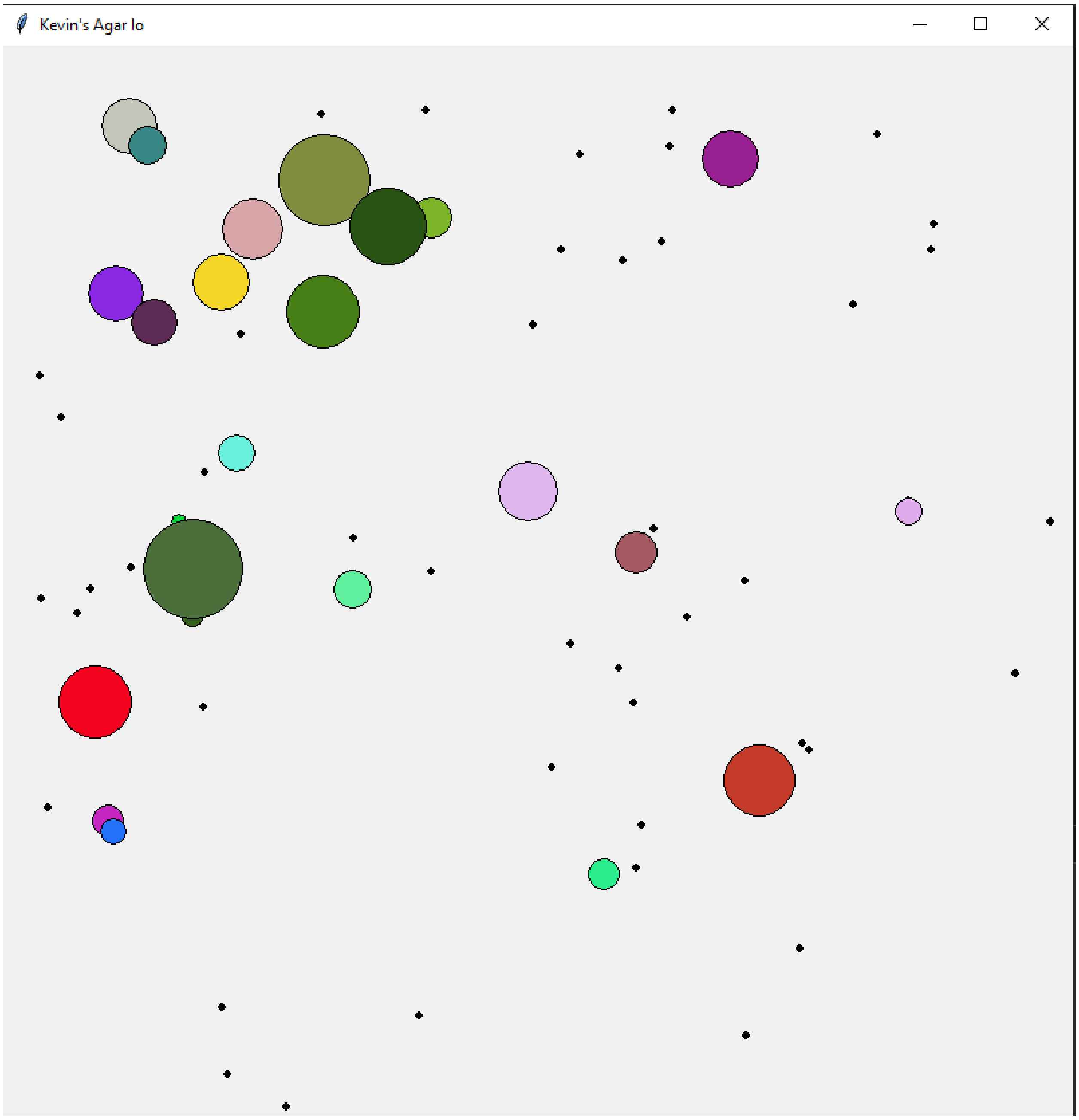
Program Name: agario_clone.py

Observations: It uses the same graphics and game library provided by Russ with no changes. Due to this, to make the program interactive, it has to access the games' private field once, which was allowed by Russ in OH.

Concept: This program uses the game and graphics library provided to construct a program that is similar to the agar.io game (https://agar.io/#ffa). This game consists of many randomly moving circles, a player circle and consumable pellets. The goal is to consume other circles while also avoiding them. The player moves using their mouse. To consume another circle, the player's circle has to be larger than the one it's trying to consume and if this has been met, it needs to simply touch/pass through a considerable amount of the enemy circle. If the consumption is successful, the player's circle is enlarged proportionally to the size of the circle it consumed. However, the opposite is also true, thus the player needs to avoid larger enemies. There is also another way to increase the size, which is less risky but also less rewarding than going after enemy circles. To do this, the player needs to consume small, static pellets that appear on the screen, though they are plenty, they provide a marginal increase in size. There are also other features in the original game such as splitting and projecting oneself, however, those have not been made in this program.

Considerations on Code: The code that allows this game to function consists of the graphics and game libraries provided by Russ. The code of the program itself constructs three classes: Player, AgarObj (for the enemies) and Pellet. Each of them have some similarities, however, they also have key differences. The player always starts with radius 20, its movement is dictated by the mouse movement and it flashes different colors for easy identification. The enemies (which are the AgarObj), move randomly in diagonal directions, have random colors (though they are static) and they can initialize anywhere on the screen and with any radius between 10 and 60. The pellets are small, static, and black circles and they are found throughout the window in random spots. To make those classes work in the game, the provided main function on the spec is used. The spawn function spawns one player object, 20 enemies and 30 pellets. The spawn_more function uses random numbers to produce chances of spawning new objects after every tick (1 in 10 to spawn new enemies and 1 in 30 to spawn 4 new pellets). There are also helper functions used to create and add those objects and they are called by both spawn and spawn_more. Another helper function used was the get_random_color, which generates a random color string for the objects. The program runs indefinitely, however, if the player is consumed at any point, the game still plays, but the player will no longer be able to interact with the window, thus it will be more of a simulation.

Example of the window's appearance:


Kevin's Agar Io