**Frogger Game Bug Report**

**Bug 1:** After successfully arriving to river part, the frog should use logs and turtles in order to go to houses. Yet, whenever the frog jumps on one of these objects, the user loses a life. The reason of this is, the game does not see the frog on those logs. It thinks that the frog fell into the water. That’s why, users can never finish a round. Instead, frogs must be able to stand on top of these logs in order to arrive their houses.

✕ on\_long = log

✓ on\_log = log

In the check\_frog\_crossing\_river() function, there was a typing mistake; therefore, the program did not run correctly. With this adjustment, the frog does not die when it stands on a turtle or a log.

**Bug 2:** If the user can not pass the round, he or she should be able to restart the level. However, whenever ‘r’ is pressed, the game does not restart. The user can not move the frog. The reason of this is that the code misses a line that runs the game once again after it is finished. Instead, the user should be able to restart the game if he or she fails a round.

✓ self.game\_running = True

I added this line at the end of the reset\_level() function to run the game once again whenever the user wants to restart. Normally, the program created everything but did not start the game so the frog could not be moved. With this addition, everything works great when the game restarts.

**Bug 3:** The frog should die after hitting a car or falling into water. That part of the game works correctly. Yet, after dying, it should return to the initial position and the user must start the round with one less life. In the game, there is a bug that prevents the frog from returning to first position after death. It starts at the same position; therefore, the frog keeps dying until the user does not have any lives remaining. Instead, the frog needs to return to the first location.

✓ self.frog.x = CANVAS\_WIDTH//2

✓ self.frog.y = GRID\_SIZE\*15

I added these lines into the died() function. These lines makes sure that the frog turns into the initial position after dying.

**Bug 4:** The primary goal of the user is to fill each house with frogs. However, the game does not allow a frog to enter the first house. Whenever a frog tries to enter, it automatically dies and the round starts again with one less life. Instead, the user should successfully put a frog into every single house.

✕ x = (spacing + GRID\_SIZE) // 2

✓ x = (spacing + GRID\_SIZE) //2 – 200

I added -200 to this calculation in the create\_homes() function. In order to find the bug, I printed coordinates of houses. I saw that even though a frog enters the first house on the screen, the x coordinate of the first house is always 200 more than the x coordinate of the frog. That’s why, the program does not see the frog in the house. Each x coordinate of houses were 200 more than the value that they should be. Therefore, I reduced 200 from every x coordinate before creating these houses. With this way, users are now able to place the frogs into each house.

**Bug 5:** Even though it is not visible on the screen, the program generates 6 houses. This does not affect the game because it is specifically stated that 5 frogs in different houses are enough to pass the round. Yet, creating six houses is still wrong, especially because that extra one can not be seen. Instead, the program should create five houses.

# in create\_homes()

✕ for i in range(0,6)

✓ for i in range(0,5)

# in reset\_homes()

✕ for i in range(0,6)

✓ for i in range(0,5)

I changed these lines in these two functions in order to create just 5 houses. Before making these adjustments, I made sure that the program created an extra house by printing each house’s x coordinate. I saw that there were 6 coordinates, which proved my idea. Again, this does not affect the game because it is directly stated in check\_frog\_entering\_home() function that five houses are enough to pass the round.

**Note:** Sometimes, the frog falls into water while walking on a log. I could not find a solution, yet it is rare and does not make game unplayable. Also, I am not sure if the last bug can be considered as a real bug. Therefore, there might be some other bugs that I did not see.