Assignment 2

Writing the C code for Frogger game

Due Date: 2024, 17 July.

1 Introduction

In this assignment, we will implement a basic C code for a well-known old game called Frogger. You can play this game online on here. To program this game we need to receive the inputs dynamically from a keyboard.

The given code <code>keyboard_input_example.c</code> is an example of how to pass inputs to a program using the keyboard. This code demonstrates how to handle keyboard inputs in a C program using low-level system calls. It continuously checks for specific key presses ('w', 'a', 's', 'd', 'q') and performs actions based on the input. The key components are:

- 1. **Header Files**: Necessary for system calls and I/O operations.
- 2. Main Function: Contains an infinite loop to continuously check for key presses.
- 3. kbhit() Function: Custom function to check if a key has been pressed.

The main function runs an infinite loop to detect and respond to specific key presses. The kbhit() function configures the terminal to non-blocking, no-echo mode, checks for a key press, and restores the terminal settings. Depending on the key pressed ('w', 'a', 's', 'd', 'q'), corresponding messages are printed or the program exits when 'q' is followed by 'y'. Try to understand and how to use it but you don't need to go into details.

2 Implementation (8 points)

The given source code [frogger.c] is a simple format of the logic you should complete. Take a look at the following code taken from [frogger.c].

```
// Game constants
#define WIDTH 64
#define HEIGHT 10

// Game global variables
int frogX, frogY;
```

This section of code defines the board structure for a simplified version of the classic game Frogger. The board is represented as a grid where certain positions are safe for the frog to move, while others are hazardous. The lanes also have speeds which determine how they move over time.

The number of columns in the board is define by WIDTH = 64, and the number of rows in the board is HEIGHT = 10. [frogX] and [frogY] represent the x-coordinate (column) and the y-coordinate (row) of the frog's position on the board, respectively.

lanes [HEIGHT] [WIDTH + 1] is a 2D array where each element is a string representing a row on the board. In each lane 'x' is hazardous positions where collisions occur and the frog cannot move (game over). On the other hand '.'s are safe positions where the frog can move. A line with all '.' indicates that the lane is entirely safe.

speeds [HEIGHT] is an array where each element corresponds to the movement speed of a lane. A positive value means the lane moves to the right, whereas then negative value shows the lane moves to the left. A lane with a zero speed remains stationary. For instance:

- 1. speeds[0] = 0: Lane 0 does not move.
- 2. speeds[1] = -2: Lane 1 moves to the left at a speed of 2 units.
- 3. speeds[2] = +1: Lane 2 moves to the right at a speed of 1 unit.

The frog starts at the bottom (lane9) and attempts to move to the top (lane0). The game checks for collisions based on the 'x' characters in the lanes array. Each lane may move left or right depending on its speed, adding dynamic obstacles for the frog.

Follow the comments within the code and implement the game's logic. At the end, your game must look like something like this example. The given format is not perfect. So you can change the format the way you want if you have a good explanation for it, and mention the reason in your report. Also, make sure you'll use only the standard library(s).

3 Report (2 points)

Explain how to compile and run you program. Create a section called Appendix and include all the codes you have in this section (Copy and paste all the codes from frogger.c into Appendix).

4 Graphical User Interface (+2 point bonus)

For this section, write all your codes in **frogger_GUI.c**. There are libraries that will allow you to create a window, render graphics, and handle user input more effectively than the current terminal-based approach. For game development, SDL2 and SFML are the most recommended due to their performance and extensive features tailored for games. This section can time consuming.

5 Submission on Avenue to Learn

- 1. The source code frogger.c.
- 2. Submit frogger_GUI.c as well, if you did the bonus part.
- 3. The Latex format of the report report.tex and the generated pdf file report.pdf.