Process of Translation of the Snake game from C to JavaScript

Seok Jun Hong

Professor Brain Howard

CSC 424A

14 December 2021

I have translated the snake game written in C to JavaScript. Through the translation process, I have encountered some difficulties. First, a program written in C does not support object-oriented principles. The program used “struct” to deliver data related to the snake node. However, it does not strongly couple methods and fields related to the snake object. I wanted to organize the code in a fine manner using object-oriented principles because it makes the program code more readable. I have used the “prototype” concept in JavaScript to implement the program using object-oriented principles. I was not used to using the prototype to define classes, so I struggled the most when dealing with the prototype.

Before starting coding, I needed to come up with the objects. Also, I have defined methods and fields which form the object. I have come up with four different prototypes, game, canvas, snake, and fruit. For each prototype, I have defined related methods and fields. For example, the snake prototype includes the “move” method that handles the movement of the snake object in the game. It also includes coordinate data that keep track of the locations of the snake object.

Using object-oriented principles to implement the program required more planning and decision-making. I had to think about which object will include the required features and organize by documenting to avoid duplicates. Also, if there is a similar feature among multiple prototypes, I had to consider extracting the feature to create a parent prototype to remove redundant codes. Since my program does not have many duplicated works, I did not need to create a parent prototype.

On the other hand, learning how HTML canvas visualization works were confusing at first. Unlike the original program written in C, JavaScript can run directly on the web browser. Therefore, I chose to use canvas in HTML to visualize the game as a website. Learning how JavaScript manipulates DOM elements took me some time, and I had to constantly check MDN Web Docs to handle web-related features like key down events.

Although I have gone through many difficulties, using JavaScript allows some advantages over C. Firstly, the object-oriented design allows us to modularize core contents in the game, which makes troubleshooting easier. When I divided the snake game into four different prototypes, I could easily see which part of the game is not functioning. Moreover, it is much easier to see the whole workflow of the program through abstraction. For example, using Snake.move() method to move the snake is very straightforward and easy to understand. Compare to the C program which took me much time to fully understand the functionality, JavaScript code is straightforward.

Secondly, JavaScript code allows us to easily implement the game on the web. Since JavaScript can run directly on the browser, we can implement the game and run it on the web without extra works. Through the web, I can easily share my snake game with other people. The original program in GitHub requires downloading the app and compiling it to test the game whereas a web-based snake game can be shared directly through a web URL. Moreover, except for the special circumstances, web-based snake game is less venerable to different user environments. Once a user has a web browser, they can enjoy the game regardless of their platform.