Abdulkareem Alqahtani

Eyad Esmail

Khalid Almanae

Dr. Christopher Martinez

System on Chips

Lab Report for Labs 6 - 8

DEC. 20 2018

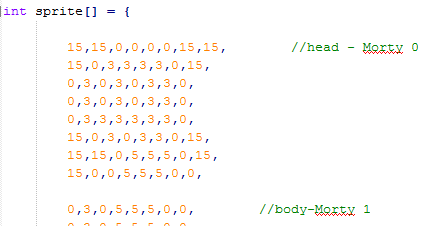
Introduction/Abstract

This is the second section in this course. It deals with writing C codes to create a GUI for the user to play the game. In the upcoming labs, we dealt with drawing sprites/elements to the screen. We had to create loops and functions that draw segments of the screen. Next, we needed to create six screens in a given order to satisfy the given criteria. Lab 7 covers moving the player. In other words, how the player moves from screen to screen, how he jumps, and how the characters interact with different aspects of the game. Lab 8 deals with enemies. The game should have had 4 types of enemies that attack the character and the player is to move around them to win.

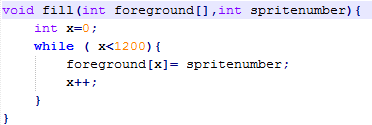
Status of overall game

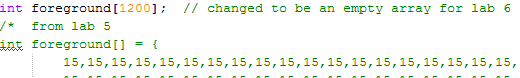
Due to time and delays from previous labs, we were only able to finish up to lab 6. We have 4 screens. The code was made for 6 screens but we have an error and we could only show 4 screens in the game. Time wasn’t in our favor. We have our characters showing as well and our controller controls the switching between screens.

Lab #6 – Drawing the screen

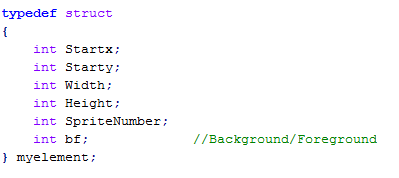
In this lab, we used what we created from lab 5 and updated the C code. Snippets will be shown with explanations. First, we converted our color pallet from names to integers. Then we added our sprites and placed them in an array. 

Next we needed to find a way to fill the foreground with the sprite. We created a function, fill, which takes in two values; the value of the position in the foreground array, and the sprite to be placed. We also emptied the arrays as we did when we placed lab 3 code in lab 5.

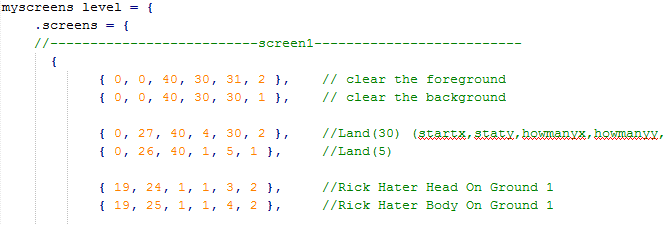




After that, we created a struct, which is our own defined type to hold several values required for the interaction. Our struct holds the start values of x and y, the width and the height we need to draw, the sprite number to be drawn and weather it is background or a foreground element.

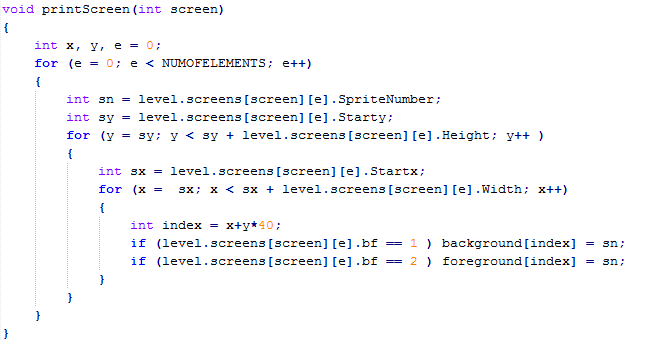


Following that, we created the screens. The screens uses the struct to place whatever data in the screen we need. The snippet bellow will show an example.

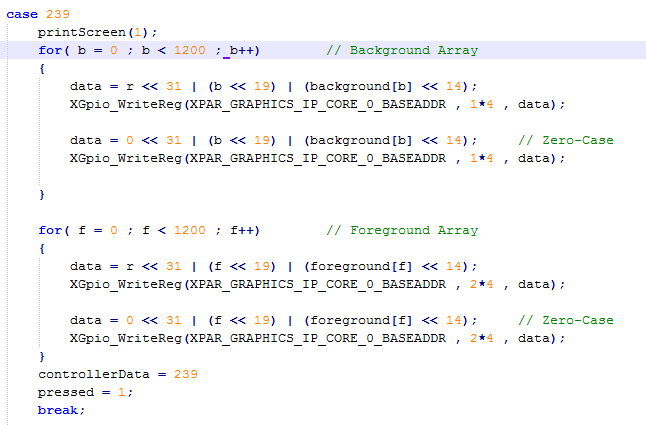


We had to place a clearing method for each screen so it doesn’t keep the objects from the previous screen in it. It was an error that was fixed.

Next we needed a print screen function to print the entire elements we have on each screen for each screen we have. To do this, we had 3 nested loops. The first loop repeats for the number of elements we have. We are using objects from the struct, the screen position and levels to be able to get what needs to be printed and where.



We used the printing loop from lab 5 with some modification and used it in each of our screens depending on what button was pressed using case statements.



This is our final progress and a video of the game running will be in the zip file.