Key Quest

Asim Rahman, Soyoung Cho

This is a simple memory game for the DE1-SoC board implemented in C

The game will generate a random letter in which the user has to type back and confirm. The next level, the number of letters the user has to type back will increase by 1. This goes on until the user makes a mistake, or they beat level 10

If at any point the user makes a mistake, its game over! The game starts back again at level 1

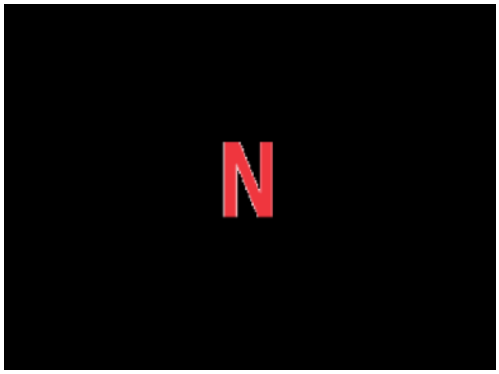
If the user is able to beat level 10, then they win!

Operating Instructions:

- At the start menu, the user can press any key to start the game

- At the beginning of each level, the level will display on the screen and HEX display

- The computer will show a series of letters



- After the computer finishes showing the letters, the user should type the letters out again one by one in the order they were shown
    - Ignore capitals (type inputs in all lowercase)
- Continue until you beat level 10 (if you can)
- If you make a mistake, no worries! Restart from level 1 and try again

Attribution Table:

(Out of 10 for each row)

Project Part	Asim	Soyoung
Game logic and states (back end)	10	
Game Visuals	1	9
Making Visual Arrays		10
Displaying Visuals/Hex display	10	
Keyboard Inputs	10	
Game Testing	3	7

Descriptions of what was done:

Asim:
I worked on the back end of the program, like the game logic and states/levels the game is in, generating the random letters to be displayed, etc… I also worked on displaying the visuals throughout the different states of the game such as starting menu, level display, letter display and game over screen. I also worked on getting the keyboard inputs from the user to check which key they pressed and comparing that input to the random letter(s) previously chosen.

Soyoung:
I mostly worked on the front end and testing the game, finding errors for this project.
I have made few versions of display components and converted into arrays and modified the array code to use it in our code. I tested the display codes to see if it displays correctly using cpulator and modified the code (display size, and the location of the images on VGA monitor).