```
// Constants
float INITIAL_FLASH_TIME = // 2.5 seconds



int flashTime = INITIAL_FLASH_TIME; // Time in milliseconds for flashing
int buttonPressCount = 0;           // Count of valid button presses
int currentLevel = 1;               // Game level

// Setup hardware interface for button press and light flashing
// enable GPIO E3 for the LEDS etc.

initializeHardware();

// Main game loop
while(true) {

   buttonPressCount = 0;


   flashGreenLight(flashTime);
   waitForButtonPress();         // wait for the button press during the flashing


   while (isButtonPressed()) {
      buttonPressCount++;
   }


   if (buttonPressCount == 0) {
      showRedLight();           '
      displayMessage("You Lose!");
      // Naro's implementation of the reset then will start the game over.

   }

   // Move to the next level after the button presses
   if (buttonPressCount > 2 && currentLevel < MAX_LEVEL)   // > than or >=  doesnt really
matter chnage easily {
      currentLevel++;
      adjustFlashSpeed();
      // subtract 1 second ???? Trial and error might be too quick but we also want a
noticable difference to show off.
      // IF we cant find a good value we can always make it very extreme for demosntration
purposes and tell this is just for show

   }
```

```
}


void initializeHardware() {

}

// Function to flash green light
void flashGreenLight(int flashDuration) {
    // Turn on green light
    turnOnGreenLight();
    delay(flashDuration);          // Wait for the flashing time duration
    turnOffGreenLight();
}


void waitForButtonPress() {
 // branch to the execption handler. if pressed i.e incrmenet cound and check if buttonCount
is > 2.
}



// Function to show red light when the player loses
void showRedLight() {
    // Flash red light indicating a loss
    turnOnRedLight();
    delay(500);  // Show red light for 500 milliseconds this is the you loose section.
    turnOffRedLight();
}


// Function to adjust flash speed based on the current level
void adjustFlashSpeed() {
 // exeption handler for the flash speeed
 // aknowledge the interupt and return.
}
```