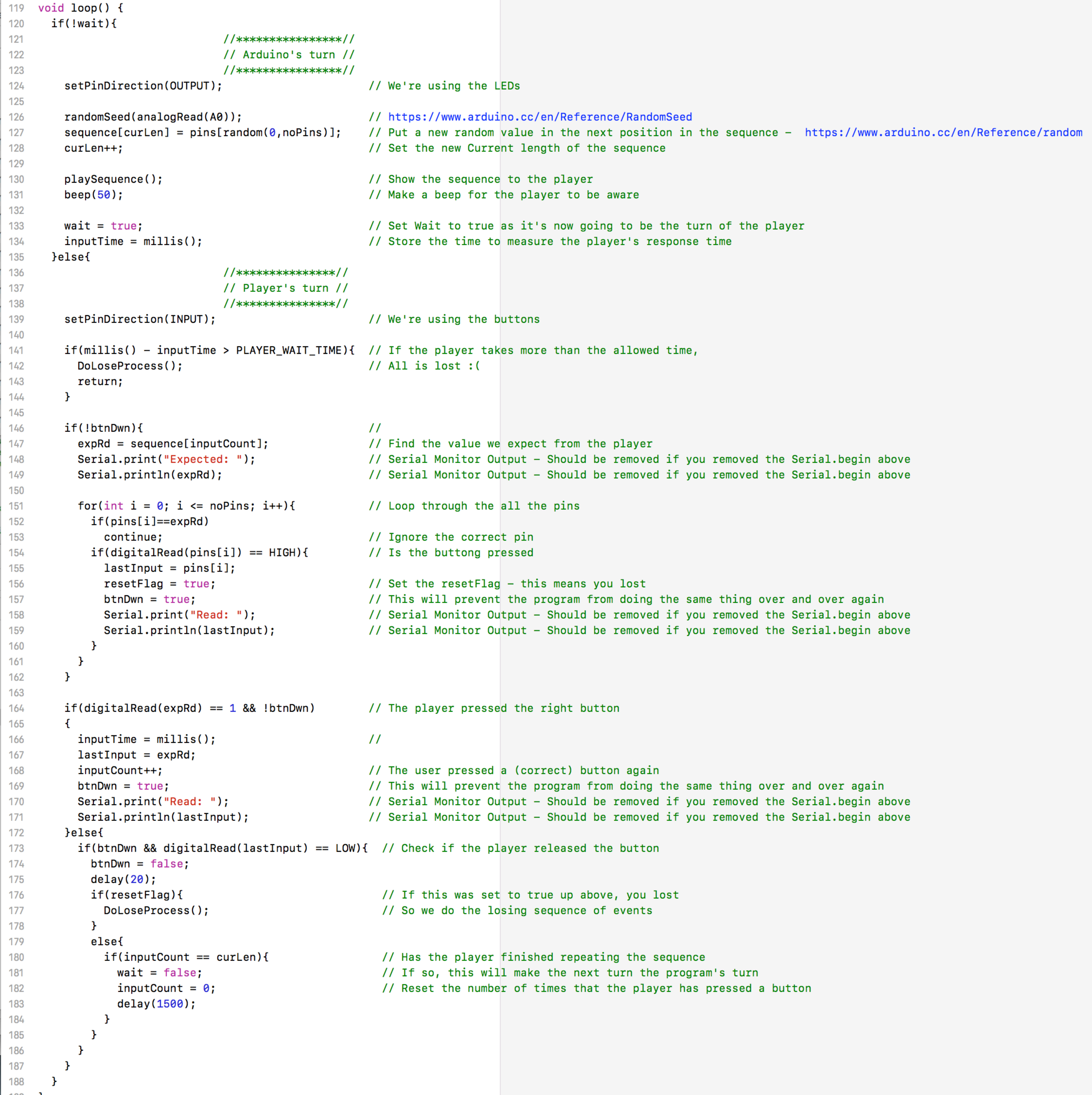

```
void loop() {
  if(!wait){
                            //****************//
                            // Arduino's turn //
                            //****************//
    setPinDirection(OUTPUT);                      // We're using the LEDs

    randomSeed(analogRead(A0));                   // https://www.arduino.cc/en/Reference/RandomSeed
    sequence[curLen] = pins[random(0,noPins)];    // Put a new random value in the next position in the sequence -  https://www.arduino.cc/en/Reference/random
    curLen++;                                     // Set the new Current length of the sequence

    playSequence();                               // Show the sequence to the player
    beep(50);                                     // Make a beep for the player to be aware

    wait = true;                                  // Set Wait to true as it's now going to be the turn of the player
    inputTime = millis();                         // Store the time to measure the player's response time
  }else{
                            //***************//
                            // Player's turn //
                            //***************//
    setPinDirection(INPUT);                       // We're using the buttons

    if(millis() - inputTime > PLAYER_WAIT_TIME){  // If the player takes more than the allowed time,
      DoLoseProcess();                            // All is lost :(
      return;
    }

    if(!btnDwn){                                  //
      expRd = sequence[inputCount];               // Find the value we expect from the player
      Serial.print("Expected: ");                 // Serial Monitor Output - Should be removed if you removed the Serial.begin above
      Serial.println(expRd);                      // Serial Monitor Output - Should be removed if you removed the Serial.begin above

      for(int i = 0; i <= noPins; i++){           // Loop through the all the pins
        if(pins[i]==expRd)
          continue;                               // Ignore the correct pin
        if(digitalRead(pins[i]) == HIGH){         // Is the buttong pressed
          lastInput = pins[i];
          resetFlag = true;                       // Set the resetFlag - this means you lost
          btnDwn = true;                          // This will prevent the program from doing the same thing over and over again
          Serial.print("Read: ");                 // Serial Monitor Output - Should be removed if you removed the Serial.begin above
          Serial.println(lastInput);              // Serial Monitor Output - Should be removed if you removed the Serial.begin above
        }
      }
    }

    if(digitalRead(expRd) == 1 && !btnDwn)        // The player pressed the right button
    {
      inputTime = millis();                       //
      lastInput = expRd;
      inputCount++;                               // The user pressed a (correct) button again
      btnDwn = true;                              // This will prevent the program from doing the same thing over and over again
      Serial.print("Read: ");                     // Serial Monitor Output - Should be removed if you removed the Serial.begin above
      Serial.println(lastInput);                  // Serial Monitor Output - Should be removed if you removed the Serial.begin above
    }else{
      if(btnDwn && digitalRead(lastInput) == LOW){  // Check if the player released the button
        btnDwn = false;
        delay(20);
        if(resetFlag){                              // If this was set to true up above, you lost
          DoLoseProcess();                          // So we do the losing sequence of events
        }
        else{
          if(inputCount == curLen){                 // Has the player finished repeating the sequence
            wait = false;                           // If so, this will make the next turn the program's turn
            inputCount = 0;                         // Reset the number of times that the player has pressed a button
            delay(1500);
          }
        }
      }
    }
  }
}
```