

```

1  #include <LiquidCrystal.h>
2
3  void setupScreen()
4  {
5      //lcd.begin(16, 2);
6      lcd.clear();
7      lcd.setCursor(0, 1);
8      lcd.write(byte(0));
9      Serial.begin(9600);
10 }
11
12 void loopScreen()
13 {
14     while(ScreenThreadFlag == 0)
15     {
16         delay(10);
17     }
18     ScreenThreadFlag = 0;
19
20     //Serial.print("(Screen Thread Wokring) ");
21
22     if(PAS!=gameOver && PAS!=Gamestart && PAS!=GameInit) // if the game is not over and
not at the starting screen
23     {
24         jumpFlag = 0;
25
26         eraseHero();
27         drawHero();
28
29         //detect collision
30         for(int i=0; i<obstcount; i++)
31         {
32             if(HeroLocation.x==obsticals[i].position.x &&
HeroLocation.y==obsticals[i].position.y)
33             {
34                 Serial.print("Game Over");
35
36                 Serial.print("Hero Position: ");
37                 Serial.print(HeroLocation.x);
38                 Serial.print(" ");
39                 Serial.println(HeroLocation.y);
40
41                 Serial.print("Obstical Position: ");
42                 Serial.print(obsticals[i].position.x);
43                 Serial.print(" ");
44                 Serial.println(obsticals[i].position.y);
45
46                 gameOva();
47             }
48         }
49     }
50
51     /*Serial.print("Sticc position: ");
52     Serial.print(sticcLocation.x);
53     Serial.print(" ");
54     Serial.println(sticcLocation.y);*/
55
56 }
57
58 void eraseHero()
59 {
60     lcd.setCursor(OldLocation.x, OldLocation.y);
61     lcd.print(" ");
62 }
63
64 void drawHero()
65 {
66     lcd.setCursor(HeroLocation.x, HeroLocation.y);
67     lcd.write(byte(0));

```

```
68     OldLocation = HeroLocation;
69 }
70
71 void eraseSticc()
72 {
73     lcd.setCursor(oldSticcLocation.x, oldSticcLocation.y);
74     lcd.print(" "); //erase the sticc
75 }
76
77 void drawSticc()
78 {
79     srand(millis());
80     if(sticcLocation.x < 0) //resetting the sticc
81     {
82         sticcLocation.x = rand()%10+8;
83     }
84
85     //increment the sticc
86     sticcLocation.x = sticcLocation.x-1;
87
88     //draw new sticc
89     lcd.setCursor(sticcLocation.x, sticcLocation.y);
90     lcd.write(byte(1));
91     /*Serial.println(sticcLocation.x);
92     Serial.println(sticcLocation.y);
93     Serial.println("Wrong");*/
94
95     oldSticcLocation = sticcLocation;
96 }
97
98 void gameOva()
99 {
100     PAS = gameOver;
101
102     lcd.clear();
103     lcd.setCursor(4, 0);
104     lcd.print("YOU DIED");
105     lcd.setCursor(4, 1);
106     lcd.print("Points: ");
107     lcd.print((points-1)); //taking out the points that got counted when the player died
108
109 }
110
```