

```
1  #include "pch.h"
2  #include "Player.h"
3  #include <iostream>
4  #include <random>
5  #include <cstdlib>
6  #include <time.h>
7  using namespace std;
8
9  #include <SFML/Graphics.hpp>
10 using namespace sf;
11
12 int x_coord, y_coord, x_move, y_move, random_int, bomb_x_coord,
    bomb_y_coord;
13 int lives = 3;
14
15 default_random_engine generator;
16 uniform_int_distribution<int> distribution(1, 20);
17
18
19 Player::Player() {
20 }
21
22 void Player::randGenerator() {
23     random_int = distribution(generator);
24 }
25
26 int** Player::p_spawn(int **g) {
27     bool valid = false;
28     while (!valid) {
29         randGenerator();
30         x_coord = random_int;
31         randGenerator();
32         y_coord = random_int;
33         cout << "Player x: " << x_coord << " y: " << y_coord << endl;
34         if (g[x_coord][y_coord] == 0)
35         {
36             g[x_coord][y_coord] = 1;
37             valid = true;
38         }
39     }
40     return g;
41 }
42
43 int Player::getX() {
44     return x_coord;
45 }
46
47 int Player::getY() {
48     return y_coord;
49 }
50
51 int** Player::move(int x, int y, int **g) {
52     x_move = x_coord + x;
```

```
53     y_move = y_coord + y;
54
55     if (g[x_coord + x][y_coord + y] == 0)
56     {
57         if (g[x_coord][y_coord] == 4) //bomb check
58         {
59             g[x_move][y_move] = 1;
60         }
61         else if (g[x_coord][y_coord] == 6) {
62             g[x_move][y_move] = 1;
63             g[x_coord][y_coord] = 4;
64         }
65         else {
66             g[x_move][y_move] = 1;
67             g[x_coord][y_coord] = 0;
68         }
69         x_coord = x_coord + x;
70         y_coord = y_coord + y;
71     }
72     return g;
73 }
74
75 void Player::minusLife() {
76     lives--;
77 }
78
79 int Player::getLives() {
80     return lives;
81 }
82
83
84 Player::~Player()
85 {
86 }
87
```