```
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;
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;
            cout << "Player x: " << x_coord << " y: " << y_coord << endl;</pre>
33
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;
```

```
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
            else if (g[x_coord][y_coord] == 6) {
61
62
                g[x_move][y_move] = 1;
63
                g[x\_coord][y\_coord] = 4;
64
            else {
65
66
                g[x_move][y_move] = 1;
                g[x\_coord][y\_coord] = 0;
67
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
```