

65010530

```

1 #include <stdio.h>
2 #include <windows.h>
3 #include <time.h>
4 #define screen_x 80
5 #define screen_y 25
6 #define scount 20
7 HANDLE rHnd;
8 HANDLE wHnd;
9 DWORD fdwMode;
10 CHAR_INFO consoleBuffer[screen_x * screen_y];
11 COORD bufferSize = { screen_x, screen_y };
12 COORD characterPos = { 0,0 };
13 SMALL_RECT windowSize = { 0, 0, screen_x - 1, screen_y - 1 };
14 COORD star[scount];
15 char ship[10] = "<-0->";
16 int life = 10, color=7;
17
18 int setConsole(int x, int y)
19 {
20     wHnd = GetStdHandle(STD_OUTPUT_HANDLE);
21     SetConsoleWindowInfo(wHnd, TRUE, &windowSize);
22     SetConsoleScreenBufferSize(wHnd, bufferSize);
23     return 0;
24 }
25
26 int setMode()
27 {
28     rHnd = GetStdHandle(STD_INPUT_HANDLE);
29     fdwMode = ENABLE_EXTENDED_FLAGS | ENABLE_WINDOW_INPUT |
30     ENABLE_MOUSE_INPUT;
31     SetConsoleMode(rHnd, fdwMode);
32     return 0;
33 }
34
35 void clear_buffer() {
36     for (int i = 0; i < screen_x * screen_y; i++) {
37         consoleBuffer[i].Char.AsciiChar = ' ';
38     }
39 }
40
41 void fill_buffer_to_console()
42 {
43     WriteConsoleOutputA(wHnd, consoleBuffer, bufferSize, characterPos,
44         &windowSize);
45 }
46
47 void init_star()
48 {
49     for (int i = 0; i < scount; i++) {
50         star[i].X = rand() % 80;
51         star[i].Y = rand() % 10;
52     }
53 }
54
55 void star_fall()
56 {
57     int i;
58     for (i = 0; i < scount; i++) {
59         if (star[i].Y == screen_y - 1) {
60             star[i].X = (rand() % 80);
61             star[i].Y = 0;
62         }
63         else {
64             star[i].Y = star[i].Y + 1;
65         }
66     }
67 }
68
69 void fill_star_to_buffer()
70 {
71     for (int i = 0; i < scount; i++) {
72         consoleBuffer[star[i].X + star[i].Y * screen_x].Char.AsciiChar = '*';
73         consoleBuffer[star[i].X + star[i].Y * screen_x].Attributes = 7;
74     }
75 }
76
77 void fill_life(int life) {
78     if (life > 9) {
79         consoleBuffer[78].Char.AsciiChar = '1';
80         consoleBuffer[78].Attributes = 7;
81         consoleBuffer[79].Char.AsciiChar = '0';
82         consoleBuffer[79].Attributes = 7;
83     }
84     else {
85         consoleBuffer[79].Char.AsciiChar = life + 48;
86         consoleBuffer[79].Attributes = 7;
87     }
88 }
89
90 void draw_ship(int x, int y) {
91     for (int i = 0; i < 5; i++) {
92         consoleBuffer[x - 2 + screen_x * y + i].Char.AsciiChar = ship[i];
93         consoleBuffer[x - 2 + screen_x * y + i].Attributes = color;
94     }
95 }
96
97 void check_collision(int x, int y) {
98     for (int i = 0; i < scount; i++) {
99         if (star[i].Y == y && star[i].X == x-2 && star[i].X == x+2) {
100             star[i].X = (rand() % 80);
101             star[i].Y = 0;
102             life -- 1;
103         }
104     }
105 }
106
107 int main()
108 {
109     int posy=25, posx=0;
110     bool play = true;
111     DWORD numEvents = 0;
112     DWORD numEventsRead = 0;
113     setConsole(screen_x, screen_y);
114     setMode();
115     srand(time(NULL));
116     init_star();
117
118     while (play)
119     {
120         GetNumberOfConsoleInputEvents(rHnd, &numEvents);
121         if (numEvents != 0)
122         {
123             INPUT_RECORD* eventBuffer = new INPUT_RECORD[numEvents];
124             ReadConsoleInput(rHnd, eventBuffer, numEvents, &numEventsRead);
125             for (DWORD i = 0; i < numEventsRead; ++i)
126             {
127                 if (eventBuffer[i].EventType == KEY_EVENT &&
128                     eventBuffer[i].Event.KeyEvent.bKeyDown == true)
129                 {
130                     if (eventBuffer[i].Event.KeyEvent.WVirtualKeyCode == VK_ESCAPE)
131                     {
132                         play = false;
133                     }
134                     else if (eventBuffer[i].Event.KeyEvent.uChar.AsciiChar == '9') {
135                         color = 1 + rand() % 9;
136                     }
137                 }
138                 else if (eventBuffer[i].EventType == MOUSE_EVENT)
139                 {
140                     posx = eventBuffer[i].Event.MouseEvent.dwMousePosition.X;
141                     posy = eventBuffer[i].Event.MouseEvent.dwMousePosition.Y;
142                     if (eventBuffer[i].Event.MouseEvent.dwButtonState &&
143                         FROM_LEFT_1ST_BUTTON_PRESSED) {
144                         color = 1 + rand() % 9;
145                     }
146                     if (eventBuffer[i].Event.MouseEvent.dwEventFlags & MOUSE_MOVED)
147                     {
148                         check_collision(posx, posy);
149                     }
150                 }
151             }
152             delete[] eventBuffer;
153         }
154         clear_buffer();
155         star_fall();
156         check_collision(posx, posy);
157         fill_star_to_buffer();
158         draw_ship(posx, posy);
159         fill_life(life);
160         fill_buffer_to_console();
161         if (life == 0) {
162             clear_buffer();
163             fill_buffer_to_console();
164             printf("GAME OVER");
165             return 0;
166         }
167         Sleep(100);
168     }
169     return 0;
170 }

```



