

ปฏิคม์ เอียวสกุลรัตน์ 65010495 Lab7 พื้นฐานการเขียน Text-mode Game ครั้งที่ 3

```
1 #include<stdio.h>
2 #include<conio.h>
3 #include<windows.h>
4 #include<stdbool.h>
5 #include<time.h>
6
7 // Set color
8 void setcolor(int fg, int bg) {
9     HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
10    SetConsoleTextAttribute(hConsole, bg * 16 + fg);
11 }
12 void gotoxy(int x, int y)
13 {
14     COORD c = { x,y };
15     SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), c);
16 }
17 // Ship
18 void draw_ship(int x, int y) {
19     gotoxy(x, y);
20     setcolor(2, 4);
21     printf("<-0->");
22 }
23 void erase_ship(int x, int y) {
24     gotoxy(x, y);
25     setcolor(0, 0);
26     printf(" ");
27 }
28 // Bullet
29 void draw_bullet(int x, int y) {
30     gotoxy(x, y);
31     setcolor(7, 1);
32     printf("^");
33 }
34 void erase_bullet(int x, int y) {
35     gotoxy(x, y);
36     setcolor(7, 0);
37     printf(" ");
38 }
39 //Star
40 void drawStar()
41 {
42     int x, y;
43     x = rand() % 71 + 10; // (10,70)
44     y = rand() % 6 + 2;   // (2,5)
45     setcolor(3, 0);
46     gotoxy(x, y);
47     printf("*");
48 }
49 //Set cursor
50 void setcursor(bool visible)
51 {
52     HANDLE console = GetStdHandle(STD_OUTPUT_HANDLE);
53     CONSOLE_CURSOR_INFO lpCursor;
54     lpCursor.bVisible = visible;
55     lpCursor.dwSize = 20;
56     SetConsoleCursorInfo(console, &lpCursor);
57 }
58 char cursor(int x, int y) {
59     HANDLE hStd = GetStdHandle(STD_OUTPUT_HANDLE);
60     char buf[2]; COORD c = { x,y };
61     DWORD num_read;
62     if (
63         !ReadConsoleOutputCharacter(hStd, (LPTSTR)buf, 1, c, (LPDWORD)&num_read))
64         return '\0';
65     else {
66         return buf[0];
67     }
68 }
69 }
70
```

```

70
71 int main()
72 {
73     setcursor(0);
74     srand(time(NULL));
75     char ch = ' ';
76     int x = 38, y = 20;
77     char direction = '0';
78     int statebullet[5] = { 0,0,0,0,0 };
79     int bulletX[5], bulletY[5];
80     int score = 0;
81     for (int i = 0; i < 20; i++)
82     {
83         drawStar();
84     }
85     setcolor(2, 0);
86     draw_ship(x, y);
87     int n = 0;
88     do {
89         if (_kbhit())
90         {
91             ch = _getch();
92             if (ch == 'a')
93             {
94                 direction = 'L'; //left
95             }
96             if (ch == 'd')
97             {
98                 direction = 'R'; //right
99             }
100             if (ch == 's')
101             {
102                 direction = 'S'; //Stop
103             }
104             //Bullet Motion
105             if (ch == ' ' && statebullet[n] == 0)
106             {
107                 Beep(700, 100);
108                 statebullet[n] = 1; //statebullet On
109                 bulletX[n] = x + 2;
110                 bulletY[n] = y - 1;
111                 n++;
112                 //reset bullet
113                 if (n == 5)
114                 {
115                     n = 0;
116                 }
117             }
118             fflush(stdin);
119         }
120         //check
121         if (direction == 'L' && x != 0)
122         {
123             erase_ship(x, y);
124             draw_ship(--x, y);
125         }
126         if (direction == 'R' && x != 80)
127         {
128             erase_ship(x, y);
129             draw_ship(++x, y);
130         }
131         if (direction == 'S' || x == 0 || x == 80)
132         {
133             draw_ship(x, y);
134         }
135         //Run statebullet
136         for (int j = 0; j < 5; j++)
137         {
138             if (statebullet[j] == 1)
139             {
140                 erase_bullet(bulletX[j], bulletY[j]); //erase current bullet
141                 if (bulletY[j] > 0 && cursor(bulletX[j], bulletY[j] - 1) != '*')
142                 {
143                     draw_bullet(bulletX[j], --bulletY[j]);
144                 }
145                 else if (cursor(bulletX[j], bulletY[j] - 1) == '*')
146                 {
147                     erase_bullet(bulletX[j], bulletY[j] - 1);
148                     drawStar();
149                     score++;
150                     Beep(700, 400);
151                     statebullet[j] = 0;
152                 }
153                 else
154                 {
155                     statebullet[j] = 0;
156                 }
157             }
158         }
159         setcolor(2, 0);
160         gotoxy(90, 3);
161         printf("Score : %d", score);
162         Sleep(100);
163     } while (ch != 'x');
164     return 0;
165 }

```

