

```

1  #include<bits/stdc++.h>
2  #include<windows.h>
3  #include<conio.h>
4  using namespace std;
5  bool gameOver;
6  const int width = 50;
7  const int height = 20;
8  int x, y, fruitX, fruitY, score;
9  int tailX[100], tailY[100];
10 int nTail;
11 enum eDirecton { STOP = 0, LEFT, RIGHT, UP, DOWN};
12 eDirecton dir;
13
14 void Setup()
15 {
16     gameOver=false;
17     dir=STOP;
18     x=width/2;
19     y=height/2;
20     fruitX=rand()%width;
21     fruitY=rand()%height;
22     score = 0;
23 }
24
25 void Draw()
26 {
27     system("cls"); //system("clear");
28     for(int i=0;i<width+2;i++)
29         cout << " ";
30     cout << endl;
31     for (int i=0;i<height;i++)
32     {
33         for (int j=0;j<width;j++)
34         {
35             if(j==0)
36                 cout<<" ";
37             if (i==y&&j==x)
38                 cout<<"O";
39             else if(i==fruitY&&j==fruitX)
40                 cout<<"B";
41             else
42             {
43                 bool print=false;
44                 for (int k=0;k<nTail;k++)
45                 {
46                     if (tailX[k]==j&&tailY[k]==i)
47                     {
48                         cout <<"o";
49                         print=true;
50                     }
51                 }
52                 if (!print)
53                     cout<<" ";
54             }
55             if (j==width-1)
56                 cout<<" ";
57         }
58         cout << endl;
59     }
60     for (int i = 0; i < width+2; i++)
61         cout << " ";
62     cout << endl;
63     cout << "Score:" << score << endl;
64     cout << endl;
65     cout << "use w,a,s,d to control the snake";
66 }
67
68 void Input()
69 {
70     if(kbhit())
71     {
72         switch (getch())
73         {
74             case 'a':
75                 dir = LEFT;
76                 break;
77             case 'd':
78                 dir = RIGHT;
79                 break;
80             case 'w':
81                 dir = UP;
82                 break;
83             case 's':
84                 dir = DOWN;

```

```

85         break;
86     case 'x':
87         gameOver = true;
88         break;
89
90     }
91 }
92
93
94 void Logic()
95 {
96     int prevX = tailX[0];
97     int prevY = tailY[0];
98     int prev2X, prev2Y;
99     tailX[0] = x;
100    tailY[0] = y;
101    for (int i=1; i<nTail; i++)
102    {
103        prev2X=tailX[i];
104        prev2Y=tailY[i];
105        tailX[i]=prevX;
106        tailY[i]=prevY;
107        prevX=prev2X;
108        prevY=prev2Y;
109    }
110    switch (dir)
111    {
112        case LEFT:
113            x--;
114            break;
115        case RIGHT:
116            x++;
117            break;
118        case UP:
119            y--;
120            break;
121        case DOWN:
122            y++;
123            break;
124        default:
125            break;
126    }
127
128    if(x>=width)
129        x=0;
130    else if(x<0)
131        x=width-1;
132
133    if(y>=height )
134        y=0;
135    else if(y<0)
136        y=height -1;
137
138    for(int i=0; i<nTail; i++)
139        if(tailX[i]==x&&tailY[i]==y)
140            gameOver=true;
141    if(x==fruitX&&y==fruitY)
142    {
143        score+=10;
144        fruitX=rand()%width;
145        fruitY=rand()%height ;
146        nTail++;
147    }
148 }
149
150 int main()
151 {
152     Setup();
153     while (!gameOver)
154     {
155         Draw();
156         Input();
157         Logic();
158         Sleep(60);
159     }
160     return 0;
161 }
162

```