```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.ComponentModel;
 4  using System.Data;
 5  using System.Drawing;
 6  using System.Linq;
 7  using System.Text;
 8  using System.Threading.Tasks;
 9  using System.Windows.Forms;
10
11  namespace FlappyBird
12  {
13      public partial class Game : Form
14      {
15          const int GraviationalConstant  = 5;
16          int gravity = GraviationalConstant;
17
18          int pipeSpeed = 10;
19
20          int score = 0;
21          int previousScore = 0;
22
23          public Game()
24          {
25              InitializeComponent();
26          }
27
28          private void gameTimer_Tick(object sender, EventArgs e)
29          {
30              bird.Top += gravity;
31
32              pipeTop.Left -= pipeSpeed;
33              pipeBottom.Left -= pipeSpeed;
34
35              if(pipeTop.Left < -90)
36              {
37                  pipeTop.Left = 700;
38                  pipeBottom.Left = 700;
39                  score++;
40                  lblScore.Text = "Score: " + score;
41                  MovePipesUpAndDown();
42              }
43
44              if(score - previousScore > 5)
45              {
46                  pipeSpeed += 10;
47                  previousScore = score;
48              }
49
50              // add pipes moving up and down
51              // I think i want to add pipes going in different directions, its
                  more fun that way
```

```
52              //I think its ok that they'll move at different heights? just make
                  differences isn't too great?
53
54              //add score increasing
55
56
57              //if (bird.Bounds.IntersectsWith(ground.Bounds)||
58              //    bird.Bounds.IntersectsWith(pipeBottom.Bounds)||
59              //    bird.Bounds.IntersectsWith(pipeTop.Bounds)||
60              //    bird.Top < -5)
61              //{
62              //    EndGame();
63              //}
64          }
65
66          private void Game_KeyDown(object sender, KeyEventArgs e)
67          {
68              if(e.KeyCode == Keys.Space)
69              {
70                  gravity = -GraviationalConstant;
71              }
72          }
73
74          private void Game_KeyUp(object sender, KeyEventArgs e)
75          {
76              if(e.KeyCode == Keys.Space)
77              {
78                  gravity = GraviationalConstant;
79              }
80          }
81
82          private void MovePipesUpAndDown()
83          {
84              Random randomNumber = new Random();
85              int verticalChange = randomNumber.Next(-80, 80);
86
87              int topPrevious = pipeTop.Top;
88              int bottomPrevious = pipeBottom.Top;
89
90              if(score % 2.0 == 0) //if score is even, move pipes up, if score is
                  odd then move pipes down
91              {
92                  pipeTop.Top += verticalChange;
93                  pipeBottom.Top += verticalChange;
94              }
95              else
96              {
97                  pipeTop.Top -= verticalChange;
98                  pipeBottom.Top -= verticalChange;
99              }
100
101             if(pipeTop.Bottom < 30) //if top pipe goes out of bounds
```

```
102             {
103                 pipeTop.Top = topPrevious;
104                 pipeBottom.Top = bottomPrevious;
105             }
106 
107             if (pipeBottom.Top > 490) //if bottom pipe goes below the ground
108             {
109                 pipeTop.Top = topPrevious;
110                 pipeBottom.Top = bottomPrevious;
111             }
112 
113         }
114 
115         private void EndGame()
116         {
117             gameTimer.Stop();
118         }
119     }
120 }
121 
```