

```
1 using System.Runtime.CompilerServices;
2 using System.Windows.Forms.VisualStyles;
3
4 namespace Sub2D
5 {
6     //Programmer: Aaron Yoon
7     //Date: 1/20/23
8     //Project: Sub2d
9     public partial class Form1 : Form
10     {
11
12         PictureBox[,] tank = new PictureBox[8, 3];
13         System.Random rY = new System.Random((int)           ↗
14             System.DateTime.Now.Ticks);
15         System.Random rX = new System.Random((int)           ↗
16             System.DateTime.Now.Ticks);
17         System.Random sX = new System.Random((int)           ↗
18             System.DateTime.Now.Ticks);
19         System.Random sY = new System.Random((int)           ↗
20             System.DateTime.Now.Ticks);
21
22         Boolean hasCrashed = false;
23
24         int pbXCoord = 3;
25         int pbYCoord = 1;
26
27         int sharkXCoord = 1;
28         int sharkYCoord = 1;
29
30         int escapeAttemptsTop = 0;
31         int escapeAttemptsBottom = 0;
32         int escapeAttemptsLeft = 0;
33         int escapeAttemptsRight = 0;
34
35         int crashes = 0;
36
37         public Form1()
38         {
39             InitializeComponent();
40         }
41
42         private void Form1_Load(object sender, EventArgs e)
43         {
44             tank[0, 0] = pic00;
45             tank[1, 0] = pic10;
46             tank[2, 0] = pic20;
```

```
46         tank[3, 0] = pic30;
47         tank[4, 0] = pic40;
48         tank[5, 0] = pic50;
49         tank[6, 0] = pic60;
50         tank[7, 0] = pic70;
51
52         tank[0, 1] = pic01;
53         tank[1, 1] = pic11;
54         tank[2, 1] = pic21;
55         tank[3, 1] = pic31;
56         tank[4, 1] = pic41;
57         tank[5, 1] = pic51;
58         tank[6, 1] = pic61;
59         tank[7, 1] = pic71;
60
61         tank[0, 2] = pic02;
62         tank[1, 2] = pic12;
63         tank[2, 2] = pic22;
64         tank[3, 2] = pic32;
65         tank[4, 2] = pic42;
66         tank[5, 2] = pic52;
67         tank[6, 2] = pic62;
68         tank[7, 2] = pic72;
69
70         tank[pbXCoord, pbYCoord].Image = picSubLeft.Image;
71         tank[sharkXCoord, sharkYCoord].Image = picShorkLeft.Image;
72
73     }
74
75     private void pictureBox1_Click(object sender, EventArgs e)
76     {
77
78     }
79
80     private void btnQuit_Click(object sender, EventArgs e)
81     {
82         this.Close();
83     }
84
85     private void btnMove_Click(object sender, EventArgs e)
86     {
87         //reset();
88         moveSub();
89         moveShark();
90         checkCollision();
91
92     }
93
94     private void moveSub()
```

```
95     {
96
97         int xMod = rX.Next(-1, 2);
98         int yMod = rY.Next(-1, 2);
99
100        //oldPos.Text = pbXCoord.ToString() + "\n" + pbYCoord.ToString()
101        //();
102
103        if ((pbXCoord == 0 && xMod == -1) || (pbXCoord == 7 && xMod ==
104        1))
105        {
106            xMod = 0;
107        }
108
109        if ((pbYCoord == 0 && yMod == -1) || (pbYCoord == 2 && yMod ==
110        1))
111        {
112            yMod = 0;
113        }
114
115        if(yMod != 0 || xMod != 0)
116            tank[pbXCoord, pbYCoord].Image = null;
117
118        pbXCoord += xMod;
119        pbYCoord += yMod;
120
121        //currentPos.Text = pbXCoord.ToString() + "\n" +
122        //pbYCoord.ToString();
123
124        if (xMod == -1)
125        {
126            tank[pbXCoord, pbYCoord].Image = picSubLeft.Image;
127        }
128        else if (xMod == 1)
129            tank[pbXCoord, pbYCoord].Image = picSubRight.Image;
130        else
131            tank[pbXCoord, pbYCoord].Image = picSubRight.Image;
132
133        if (pbXCoord == 0)
134            escapeAttemptsLeft++;
135        if (pbXCoord == 7)
136            escapeAttemptsRight++;
137        if(pbYCoord == 0)
138            escapeAttemptsBottom++;
139        if(pbYCoord == 2)
```

```
140         escapeAttemptsTop++;
141
142         subPos.Text = pbXCoord.ToString() + pbYCoord.ToString();
143         checkCollision();
144     }
145
146     public void reset()
147     {
148         // if (hasCrashed)
149         //{
150             tank[0, 0].Image = null;
151             tank[1, 0].Image = null;
152             tank[2, 0].Image = null;
153             tank[3, 0].Image = null;
154             tank[4, 0].Image = null;
155             tank[5, 0].Image = null;
156             tank[6, 0].Image = null;
157             tank[7, 0].Image = null;
158
159             tank[0, 1].Image = null;
160             tank[1, 1].Image = null;
161             tank[2, 1].Image = null;
162             tank[3, 1].Image = null;
163             tank[4, 1].Image = null;
164             tank[5, 1].Image = null;
165             tank[6, 1].Image = null;
166             tank[7, 1].Image = null;
167
168             tank[0, 2].Image = null;
169             tank[1, 2].Image = null;
170             tank[2, 2].Image = null;
171             tank[3, 2].Image = null;
172             tank[4, 2].Image = null;
173             tank[5, 2].Image = null;
174             tank[6, 2].Image = null;
175             tank[7, 2].Image = null;
176
177             sharkXCoord = 0;
178             sharkYCoord = 1;
179
180             pbXCoord = 7;
181             pbYCoord = 1;
182
183             hasCrashed = false;
184
185             tank[sharkXCoord, sharkYCoord].Image =
186                 picShorkRight.Image;
187             tank[pbXCoord, pbYCoord].Image = picSubLeft.Image;
188         //}
```

```
188     }
189     public void moveShark()
190     {
191         int xMod = rX.Next(-1, 2);
192         int yMod = rY.Next(-1, 2);
193
194         //oldPos.Text = pbXCoord.ToString() + "\n" + pbYCoord.ToString()
195         ();
196
197         if ((sharkXCoord == 0 && xMod == -1) || (sharkXCoord == 7 &&
198             xMod == 1))
199         {
200             xMod = 0;
201         }
202
203         if ((sharkYCoord == 0 && yMod == -1) || (sharkYCoord == 2 &&
204             yMod == 1))
205         {
206             yMod = 0;
207         }
208
209         if (yMod != 0 || xMod != 0)
210             tank[sharkXCoord, sharkYCoord].Image = null;
211
212         sharkXCoord += xMod;
213         sharkYCoord += yMod;
214
215         //currentPos.Text = pbXCoord.ToString() + "\n" +
216         pbYCoord.ToString();
217
218         if (xMod == -1)
219         {
220             tank[sharkXCoord, sharkYCoord].Image = picShorkLeft.Image;
221         }
222         else if (xMod == 1)
223             tank[sharkXCoord, sharkYCoord].Image =
224             picShorkRight.Image;
225         else
226             tank[sharkXCoord, sharkYCoord].Image =
227             picShorkRight.Image;
228
229         sharkPos.Text = sharkXCoord.ToString() + sharkYCoord.ToString()
230         ();
231         checkCollision();
232     }
```

```
230
231     public void checkCollision()
232     {
233         if(sharkXCoord == pbXCoord && sharkYCoord == pbYCoord)
234         {
235             tank[sharkXCoord, sharkYCoord].Image = null;
236             tank[sharkXCoord, 0].Image = picBoom.Image;
237             crashes++;
238             //sharkXCoord = 0;
239             //sharkYCoord = 1;
240             //pbXCoord = 7;
241             //pbYCoord = 1;
242             //MessageBox.Show("Game Over!");
243             hasCrashed = true;
244             timer1.Enabled = false;
245         }
246     }
247
248
249
250     private void btnAutomate_Click(object sender, EventArgs e)
251     {
252         if (timer1.Enabled)
253             timer1.Enabled = false;
254         else if (!timer1.Enabled)
255             timer1.Enabled = true;
256     }
257
258     private void txtInterval_TextChanged(object sender, EventArgs e)
259     {
260         try
261         {
262             timer1.Interval = int.Parse(txtInterval.Text);
263         }
264
265         catch
266         {
267             MessageBox.Show("Please put an integer");
268         }
269     }
270
271
272     private void timer1_Tick(object sender, EventArgs e)
273     {
274         //reset();
275         moveSub();
276         moveShark();
277         checkCollision();
278     }
```

```
279     }
280
281     private void btnStats_Click(object sender, EventArgs e)
282     {
283         MessageBox.Show(
284             "top bumps: " + escapeAttemptsTop.ToString() + "\n" +
285             "bottom bumps: " + escapeAttemptsBottom.ToString() + "\n" +
286             "left bumps: " + escapeAttemptsLeft.ToString() + "\n" +
287             "right bumps: " + escapeAttemptsRight.ToString() + "\n" +
288             "crashes: " + crashes.ToString()
289         );
290     }
291
292     private void btnReset_Click(object sender, EventArgs e)
293     {
294         reset();
295     }
296 }
297 }
```