

```
1 using System;
2
3 namespace AtomicGroundStateEnergiesGaussian
4 {
5     class Integration
6     {
7         // Multidimensional Simpson's Rule
8         public double Integrate(
9             double[] lower, double[] upper, double[] alpha,
10             Func<double[], double[], double> f, int n, int[] steps)
11         {
12             double p = 1;
13             double[] h = new double[n];
14             double[] h2 = new double[n];
15             double[] s = new double[n];
16             double[] t = new double[n];
17             double[] x = new double[n];
18             double[] w = new double[n];
19
20             for (int i = 0; i < n; i++)
21             {
22                 h[i] = (upper[i] - lower[i]) / steps[i];
23                 h2[i] = 2.0 * h[i];
24             }
25
26             for (int i = 0; i < n; i++)
27             {
28                 for (int j = 0; j < n; j++)
29                     x[j] = lower[j];
30
31                 x[i] = lower[i] + h[i];
32
33                 for (int j = 1; j < steps[i]; j += 2)
34                 {
35                     s[i] += f(x, alpha);
36                     x[i] += h2[i];
37                 }
38
39                 for (int j = 0; j < n; j++)
40                     x[j] = lower[j];
41
42                 x[i] = lower[i] + h2[i];
43
44                 for (int j = 2; j < steps[i]; j += 2)
45                 {
46                     t[i] += f(x, alpha);
47                     x[i] += h2[i];
48                 }
49
50                 w[i] = h[i] * (f(lower, alpha) + 4 * s[i] +
51                     2 * t[i] + f(upper, alpha)) / 3.0;
52             }
53         }
54     }
55 }
```

```
53
54         for (int i = 0; i < n; i++)
55             p *= w[i];
56
57         return p;
58     }
59 }
60 }
```