```
1  using System;
2  using System.IO;
3
4  namespace Algorithms
5  {
6
7      public class Erathostenes
8      {
9          public static void Main()
10         {
11             new Erathostenes().Run(Console.In, Console.Out);
12         }
13
14         public void Run(TextReader reader, TextWriter writer)
15         {
16             int n = int.Parse(reader.ReadLine());
17             var results = Sieve(n);
18
19             foreach (int result in results)
20             {
21                 if (result > 0)
22                     writer.Write(result + "\t");
23             }
24
25             reader.ReadLine();
26
27         }
28         /// <summary>
29         /// Implementes the sieve of Eratosthenes
30         /// Input: An integrer n>=2
31         /// Output: Array L of all prime numbers less than or equal to n
32         /// </summary>
33         public Array Sieve(int n)
34         {
35             int[] A = new int[n + 2];
36             for (int p = 2; p <= n; p++)
37                 A[p] = p;
38
39             for (int p = 2; p <= Math.Floor(Math.Sqrt(n)); p++)
40             {
41                 if (A[p] != 0)
42                 {
43                     int j = p * p;
44                     while (j <= n)
45                     {
46                         A[j] = 0;
47                         j = j + p;
48                     }
49                 }
50             }
51             int i = 0;
52             int[] L = new int[A.Length];
53             for (int p = 2; p <= n; p++)
54             {
55                 if (A[p] != 0)
56                 {
```

```
57                      L[i] = A[p];
58                      i++;
59                  }
60              }
61              return L;
62          }
63      }
64  }
65
```