```
 1  using System;
 2  using System.Linq;
 3  using System.Text;
 4
 5  namespace LabAssignment1
 6  {
 7
 8      class IntegerSet
 9      {
10          bool[] boolArray;
11          private const int MAXSIZE = 101;
12
13          /// <summary>
14          /// Initialize all the elements of the array with the default value -
              FALSE.
15          /// </summary>
16          public IntegerSet()
17          {
18              boolArray = new bool[MAXSIZE];
19          }
20
21          /// <summary>
22          /// Initialize all the elements of the array with the default value -
              FALSE.
23          /// Assign the value of the arrays element to TRUE if it the given
              input falls in the accepted range.
24          /// </summary>
25          /// <param name="intArray"></param>
26          public IntegerSet(int[] intArray)
27          {
28              // Initialize all the elements of the array with the default value
                - FALSE.
29              boolArray = new bool[MAXSIZE];
30              foreach (int element in intArray)
31              {
32                  if (element < MAXSIZE && element >= 0)
33                      // Assign the value of the arrays element to TRUE if it the
                         given input falls in the accepted range.
34                      boolArray[element] = true;
35              }
36          }
37
38          /// <summary>
39          /// Assign the input array to the IntegerSet's boolArray.
40          /// </summary>
41          /// <param name="bArray"></param>
42          public IntegerSet(bool[] bArray)
43          {
44              // Assign the input array to the IntegerSet's boolArray.
```

```
45              this.boolArray = bArray;
46          }
47
48          /// <summary>
49          /// Set the value of the boolArray's element to TRUE - to indicate that
                the new element is added to the set.
50          /// </summary>
51          /// <param name="e"></param>
52          public void InsertElement(int e)
53          {
54              // Set the value of the boolArray's element to TRUE  - to indicate
                  that the new element is added to the set.
55              if(e<101 && e>0)
56                  boolArray[e] = true;
57          }
58
59          /// <summary>
60          /// Set the value of the boolArray's element to FALSE  - to indicate
                that the new element is deleted from the set.
61          /// </summary>
62          /// <param name="e"></param>
63          public void DeleteElement(int e)
64          {
65              // Set the value of the boolArray's element to FALSE  - to indicate
                  that the new element is deleted from the set.
66              boolArray[e] = false;
67          }
68
69          /// <summary>
70          /// To check if both the input sets are equal or not.
71          /// </summary>
72          /// <param name="obj"></param>
73          /// <returns> Boolean value based on the input set. </returns>
74          public bool IsEqualTo(IntegerSet obj)
75          {
76              IntegerSet iSet = (IntegerSet)obj;
77              for (int i = 0; i < MAXSIZE; i++)
78              {
79                  if (this.boolArray[i] != iSet.boolArray[i])
80                      // If any of the elements in both the sets are not equal,
                        then return false
81                      return false;
82              }
83              return true;
84          }
85
86          /// <summary>
87          /// Returns a string containing a set as a list of numbers separated by
                spaces.
```

```
88          /// Only those elements that are present in the set are displayed.
89          /// Empty set is represented by "---".
90          /// </summary>
91          /// <returns> Returns a String containing elements that are present in
              the set. If empty then "---". </returns>
92          public string ToString()
93          {
94              string a = default;
95              int iFalseCounter = 0;
96              for(int i=0;i<this.boolArray.Length;i++)
97              {
98                  if (this.boolArray[i] == true)
99                      // Whenever the element value is TRUE, we append the value
                        of a with the index of that element.
100                     a += i + " ";
101                 else
102                 {
103                     // Whenever the element value is FALSE, we increase the
                        counter of iFalseCounter.
104                     iFalseCounter++;
105                 }
106             }
107             // If the value of iFalseCounter is equal to the number of elements
                   in the set, it means that the set is empty.
108             if (iFalseCounter==this.boolArray.Length)
109                 // Return the value of a as "---"
110                 return a = "---";
111             else
112                 // If there are any valid elements, then return the index of
                      those elements.
113                 return a;
114         }
115
116         /// <summary>
117         /// Union of two input sets are computed and stored in new set.
118         /// </summary>
119         /// <param name="obj"></param>
120         /// <returns> Returns IntegerSet object </returns>
121         public IntegerSet Union(IntegerSet obj)
122         {
123             // New set - tempArray is created with the default value - FALSE.
124             bool[] tempArray = new bool[MAXSIZE];
125             for(int i = 0; i < MAXSIZE; i++)
126             {
127                 // Store the value of the new set based on the output of OR of
                      the two input sets.
128                 tempArray[i] = this.boolArray[i] || obj.boolArray[i];
129             }
130             // Get the corresponding IntegerSet object.
```

```
131             IntegerSet iSet = new IntegerSet(tempArray);
132             return iSet;
133         }
134 
135         /// <summary>
136         /// Intersection of two input sets are computed and stored in new set.
137         /// </summary>
138         /// <param name="obj"></param>
139         /// <returns> Returns IntegerSet object </returns>
140         public IntegerSet Intersection(IntegerSet obj)
141         {
142             // New set - tempArray is created with the default value - FALSE.
143             bool[] tempArray = new bool[MAXSIZE];
144             for (int i = 0; i < MAXSIZE; i++)
145             {
146                 // Store the value of the new set based on the output of AND of
                       the two input sets.
147                 tempArray[i] = this.boolArray[i] && obj.boolArray[i];
148             }
149             // Get the corresponding IntegerSet object.
150             IntegerSet iSet = new IntegerSet(tempArray);
151             return iSet;
152         }
153     }
154 }
155 
```