```csharp
using System;
using System.Collections.Generic;
public static class SumOfMultiples
{
    public static int Sum(IEnumerable<int> multiples, int max)
    {
        int result;
        int count = 2;
        int totalSum;

        List<int> listMultiple = new List<int>();
        List<int> listSort = new List<int>();
        List<int> listWODuplicates = new List<int>();
        foreach (int multiple in multiples)
        {
            if (multiple == 0)
            {
                listMultiple.Add(multiple);
                break;
            }

            result = multiple;
            while (result < max)
            {
                listMultiple.Add(result);
                result = multiple * count;
                count++;
            }
            count = 2;
        }
        listSort = SortList(listMultiple);
        listWODuplicates = RemoveDuplicates(listSort);
        totalSum = CalculateSum(listWODuplicates);

        return totalSum;
    }


    // Funcion que ordena la lista de menor a mayor
    public static List<int> SortList(List<int> list)
    {
        int aux;
        for (int i = 0; i < list.Count - 1; i++)
        {
            for (int j = i + 1; j < list.Count; j++)
            {
                if (list[i] > list[j])
                {
                    aux = list[i];
                    list[i] = list[j];
                    list[j] = aux;
                }
            }
```

```csharp
54              }
55          return list;
56      }
57
58
59      // Funcion que elimina los valores reptidos
60      public static List<int> RemoveDuplicates(List<int> list)
61      {
62          for (int i = 0; i < list.Count - 1; i++)
63          {
64              for (int j = i + 1; j < list.Count; j++)
65              {
66                  if (list[i] == list[j])
67                  {
68                      list.RemoveAt(j);
69                      j--;
70                  }
71              }
72          }
73          return list;
74      }
75
76
77      // Funcion que calcula el sumatorio de los elementos de la lista
78      public static int CalculateSum(List<int> list)
79      {
80          int result = 0;
81          foreach (int i in list)
82          {
83              result += i;
84          }
85          return result;
86      }
87 }
```