sum_NOn_Common_Elements

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;



namespace Practiceall
{
    internal class Usercode4
    {
        public static int NonCommonElements(int n1, int n2, int[] a, int[] b)
        {
            //write code here
            int sum1 = 0, sum2 = 0, result, temp1 = 0, temp2 = 0;
            for (int i = 0; i < n1; i++)
            {
                if (a[i] < 0)
                {
                    temp1++;
                }
            }
            for (int i = 0; i < n2; i++)
            {
                if (b[i] < 0)
                {
                    temp2++;
                }
            }
            if (temp1 != 0 && temp2 != 0)
            {
                Console.WriteLine("Input1 and Input2 has negative numbers");
                return -3;
            }
            else if (temp1 != 0)
            {
                Console.WriteLine("Input1 has negative numbers");
                return -1;
            }
            else if (temp2 != 0)
            {
                Console.WriteLine("Input2 has negative numbers");
                return -2;
```

```csharp
            }
            else
            {
                int[] arr1 = a.Intersect(b).ToArray();
                int[] arr2 = a.Union(b).ToArray();
                for (int i = 0; i < arr1.Length; i++)
                {
                    sum1 = sum1 + arr1[i];
                }
                for (int i = 0; i < arr2.Length; i++)
                {
                    sum2 = sum2 + arr2[i];
                }
                result = sum2 - sum1;
                Console.WriteLine("Sum of Noncommon elements is {0}", result);
                return result;
            }
        }
    }
    public class Program4
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Enter array size for first array ");
            int n1 = int.Parse(Console.ReadLine());
            Console.WriteLine("Enter array size for second array");
            int n2 = int.Parse(Console.ReadLine());
            int[] a = new int[n1];
            int[] b = new int[n2];
            Console.WriteLine("Enter values for first array");
            for (int i = 0; i < n1; i++)
            {
                a[i] = int.Parse(Console.ReadLine());
            }
            Console.WriteLine("Enter values for second array");
            for (int i = 0; i < n2; i++)
            {
                b[i] = int.Parse(Console.ReadLine());
            }
            int res = Usercode4.NonCommonElements(n1, n2, a, b); Console.ReadLine();
        }
    }
}
```

calculate cost

```csharp
namespace flower
{
```

```
class UserMainCode
{
public static int fun(int cost, char dtype, char ftype)
{
int total;
if (dtype == 'S')
{
if (ftype == 'N')
{
total = 15000 + (cost * 400);
}
else if (ftype == 'E')
{
total = 15000 + (cost * 700);
}
else
{
total = -2; return total;
}
}
else if (dtype == 'C')
{
if (ftype == 'N')
{
total = 25000 + (cost * 400);
}
else if (ftype == 'E')
{
total = 25000 + (cost * 700);
}
else
{
total = -2; return total;
}
}
else
{
total = -3; return total;
}
if (total < 20000)
{
total = -1;
}
return total;
}
}
```

Another Method:

```
public static void count(int n,char a,char b)
{
int flower; if (a == 'n')
{
flower = n * 400;
}
else
{
flower = n * 700;
}
if (b == 's')
{
flower = flower + 15000;
}
else
{
flower = flower + 25000;
}
if (flower < 20000)
{
Console.WriteLine(-1);
}
else
{
Console.WriteLine(flower);
}
}
```

Telephone Bills:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace akhil
{
    class userprogramcode_Telephone
    {
        public static double user(int r)
        {
            double b = 0.00;
            if (r <= 300)
            {
                b = 200;
            }
```

```
        else if (r > 300 && r <= 350)
        {
            b = Convert.ToDouble((r - 300) * 0.60 + 200);
        }
        else if (r > 350 && r <= 400)
        {
            b = Convert.ToDouble((r - 350) * 0.50 + (50 * 0.60) + 200);
        }
        else
        {
            b = Convert.ToDouble((r - 400) * 0.40 + (50 * 0.50) + (50 * 0.60) + 200);
        }
        double output = Math.Round(b, 2);
        return output;
    }
}

class Program_TelephoneBill
{
    public static void Main()
    {
        int r = int.Parse(Console.ReadLine());
        double output = userprogramcode_Telephone.user(r);
        Console.WriteLine(output.ToString("0.00"));
    }
}
}
```

Find the Average:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication18
{
    class UserProgramCode
    {
        public static float compute(int[] array, int size)
        {
            float avg, sum = 0; int i;
            foreach (int a in array)
            {
                if (a < 0) return -1;
            }
            for (i = 0; i < size; i++)
            {
                sum = sum + array[i];
```

```csharp
        }
        avg = sum / size;
        return avg;
    }
}

class Program
{
    static void Main(string[] args)
    {
        UserProgramCode u = new UserProgramCode(); int n;
        float avg;
        n = int.Parse(Console.ReadLine()); int[] a = new int[n];
        for (int i = 0; i < n; i++)
        {
            a[i] = int.Parse(Console.ReadLine());
        }
        avg = UserProgramCode.compute(a, n); if (avg == -1)
        {
            Console.WriteLine("Negative numbers present");
        }
        else Console.WriteLine(String.Format("{0:0.0}", avg));
    }
}
}
```

another method:
```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace akhil
{
    class userprogramcode_Average
    {
        public static double Average(int[] input1, int n)
        {
            float sum = 0;
            for (int i = 0; i < n; i++)
            {
                sum = sum + input1[i];
            }
            return sum / n;
        }
    }
```

```csharp
class Program_AVverage
{
    public static void Main()
    {
        int n = int.Parse(Console.ReadLine());
        int[] input1 = new int[n];
        for (int i = 0; i < n; i++)
        {
            input1[i] = int.Parse(Console.ReadLine());
        }
        double result = userprogramcode_Average.Average(input1, n);
        Console.WriteLine(result);
    }
}
}
```

Find nth Largest Number

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace akhil
{
    class userprogramcode_Nth_LargestElement
    {
        public static int NthLargest(int[] input1)
        {
            int count = input1.Length;
            int count1 = 0;
            int output;
            for (int i = 0; i < count; i++)
            {
                if (input1[i] > 0)
                {
                    count1++;
                }
            }
            if (count == count1)
            {
                Array.Sort(input1);
                Array.Reverse(input1);
                output = input1[0];
            }
            else
            {
```

```
        output = -1;
    }
        return output;
    }
  }

  class Program_Nth_largest_element
  {
    public static void Main()
    {

      int n = int.Parse(Console.ReadLine());
      int[] inpt1 = new int[n];
      for (int i = 0; i < n; i++)
      {
        inpt1[i] = int.Parse(Console.ReadLine());
      }
      int result = userprogramcode_Nth_LargestElement.NthLargest(inpt1);
      Console.WriteLine(result);
    }
  }
}
```

Image type:

```
public static void user(string[] st,int n)
{
List<string> l=new List<string>();
List<string> l1 = new List<string>();
 int count=0;
string[] st2 = { "jpeg", "jfif","exif","tiff","raw","gif","bmp","png" };
string[] st1=new string[2];
foreach (var item in st)
{
st1=item.Split('.');
for (int i = 0; i < 2; i++)
{
l.Add(st1[i]);
}
}

for (int i = 0; i < 8; i++)
{
count = 0;
foreach (var item in l)
{
if (st2[i] == item)
{
```

```csharp
count++;
}
}
if (count != 0)
{
l1.Add(st2[i]); l1.Add(count.ToString());
}

}
foreach (var item in l1)
{
Console.WriteLine(item);
}
```