BDAT 1004 – Data Programming

Problem Set - 3

Question 1 C#

A Data Programming professor has a class of students. Frustrated with their lack of discipline, he decides to cancel class if fewer than some number of students are present when class starts. Arrival times go from on time ($arrivalTime \leq 0$) to arrived late ($arrivalTime > 0$).

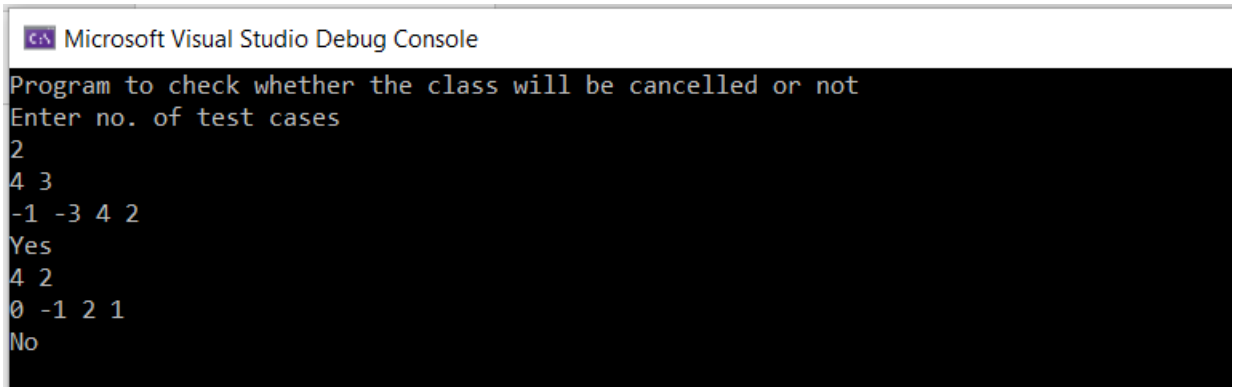
Given the arrival time of each student and a threshold number of attendees, determine if the class is canceled.

Program.cs

```
using System;

namespace AngryProfessor
{
    class Program
    {
        public static object readLine { get; private set; }

        static void Main(string[] args)
        {
            Console.WriteLine("Program to check whether the class will be cancelled or
not");

            Console.WriteLine("Enter no. of test cases");
            int test_cases = Convert.ToInt32(Console.ReadLine());
            if (test_cases < 1 && test_cases > 10)
            {
                Console.WriteLine("Test case value should be 1 to 10");
                Console.WriteLine("in else");
                System.Environment.Exit(1);
            }
            else
            {
                for (int i = 0; i < test_cases; i++)
                {

                    string[] tokens = Console.ReadLine().Split();
                    int no_of_student = int.Parse(tokens[0]);
                    //Console.WriteLine(no_of_student);
                    int cancellation_threshold = int.Parse(tokens[1]);
                    //Console.WriteLine(cancellation_threshold);

                    string readLine = Console.ReadLine();
                    string[] stringArray = readLine.Split(' ');

                    //Console.WriteLine(stringArray);
                    int[] a = new int[stringArray.Length];
                     //Console.WriteLine(a);
```

```
                    for (int j = 0; j < stringArray.Length; j++)
                    {
                        a[j] = int.Parse(stringArray[j]);
                    }

                     angryprofessor(cancellation_threshold, a);
                }
            }
        }

        public static void angryprofessor(int cancellation_threshold, int[] a)
        {
            int count = 0;
            for (int i=0;i<a.Length; i++)
            {
                if(a[i]<1 || a[i]==0)
                {
                    count = count + 1;
                }
            }
            if (count >= cancellation_threshold)
            {
                Console.WriteLine("No");
            }
            else
            {
                Console.WriteLine("Yes");
            }
        }
    }
}
```

Output:



Question 2 C#

You will be given two arrays of integers and asked to determine all integers that satisfy the following two conditions:
1. The elements of the first array are all factors of the integer being considered

2. The integer being considered is a factor of all elements of the second array

Program.cs

```
using System;

namespace gettotal_question_2
{
    class Program
    {
        static void Main()
        {
            string[] tokens = Console.ReadLine().Split();
            int n = int.Parse(tokens[0]);
            int m = int.Parse(tokens[1]);
            //Console.WriteLine(n);
            //Console.WriteLine(m);

            string a = Console.ReadLine();
            string[] StrArray_a = a.Split(' ');


            int[] array_a = new int[StrArray_a.Length];
            //Console.WriteLine(a);
            for (int j = 0; j < StrArray_a.Length; j++)
            {
                array_a[j] = int.Parse(StrArray_a[j]);
            }

            string b = Console.ReadLine();
            string[] StrArray_b = b.Split(' ');

            int[] array_b = new int[StrArray_b.Length];
            //Console.WriteLine(b);
            for (int j = 0; j < StrArray_b.Length; j++)
            {
                array_b[j] = int.Parse(StrArray_b[j]);
            }

            GetTotalX(array_a, array_b);
        }


        public static void GetTotalX(int[] array_a, int[] array_b)
        {

            int x = 1, r = 0, j = 0, count = 0;
            int[] d = new int[100];
```

```
            for (x = 1; x < 100; x++)
            {
                int c = 0;
                for (int i = 0; i < array_a.Length; i++)
                {
                    if (x % array_a[i] == 0 && x >= array_a[i])
                    {
                        c++;
                    }
                }
                if (c == array_a.Length)
                {
                    d[j] = x;
                    r++;
                    j++;
                }
            }
            for (j = 0; j < r; j++)
            {
                int c = 0;
                for (int i = 0; i < array_b.Length; i++)
                {
                    if (array_b[i] % d[j] == 0)
                    {
                        c++;
                    }
                }
                if (c == array_b.Length)
                {
                    count++;
                }
            }
            Console.WriteLine(count);
        }
    }
}
```

Output: