



Federal Urdu University
of Arts, Science and Technology

FEDERAL URDU UNIVERSITY OF ARTS SCIENCE AND TECHNOLOGY

ELEMENTARY EXERCISES

“ ASSIGNMENT NO 1 ”

UZAIR AHMED

SEAT NO: 19122126

VISUAL PROGRAMMING

Professor Name: Mr. Muhammad Tauseef



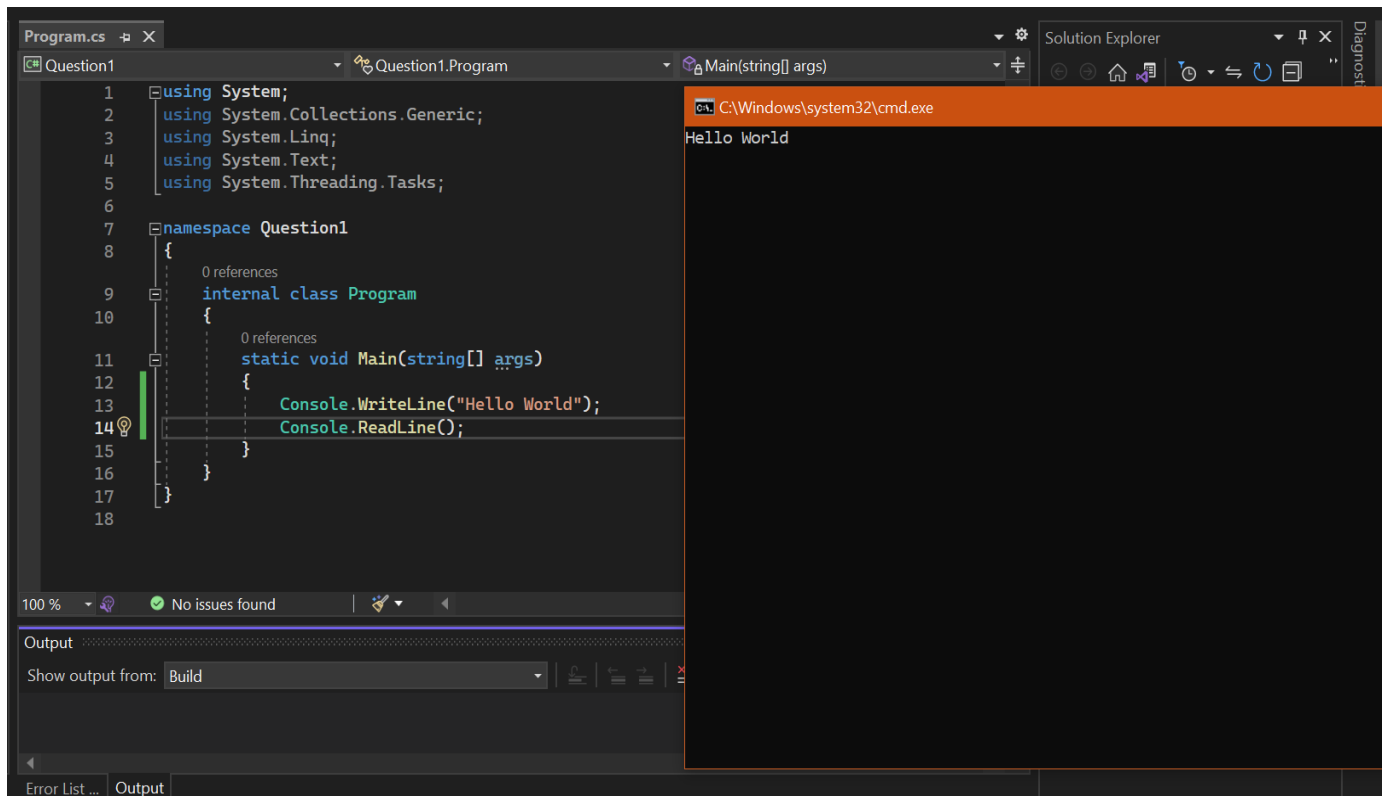
Assignment Due Date

April 09, 2023

Q1. Write a program that prints 'Hello World' to the screen.

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

namespace Question1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("HELLO WORLD");
            Console.ReadLine();
        }
    }
}
```

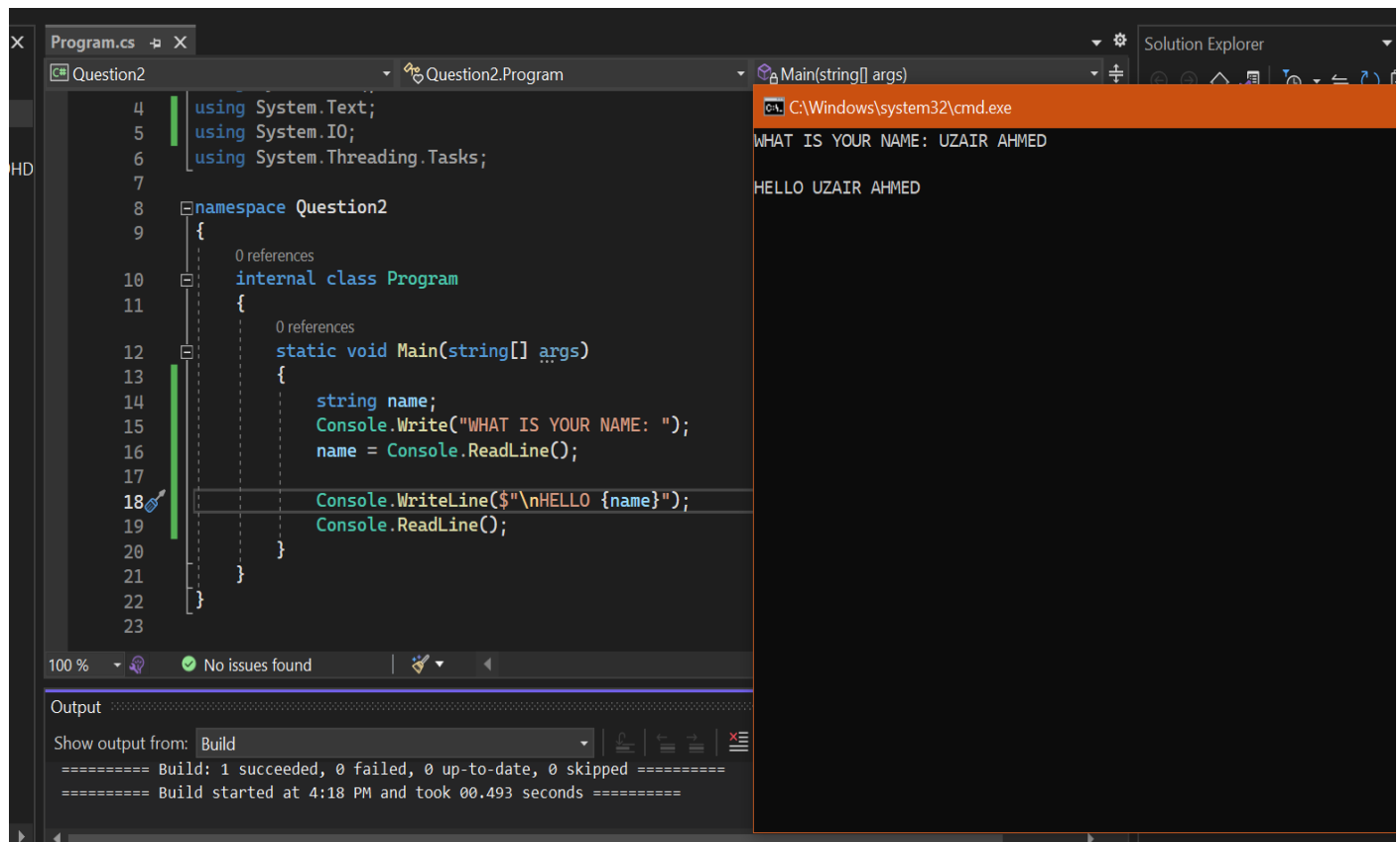


Q2. Write a program that asks the user for their name and greets them with their name.

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

namespace Question2
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string name;
            Console.Write("WHAT IS YOUR NAME: ");
            name = Console.ReadLine();

            Console.WriteLine($"\\nHELLO {name}");
            Console.ReadLine();
        }
    }
}
```



Q3. Modify the previous program such that only the users Alice and Bob are greeted with their names

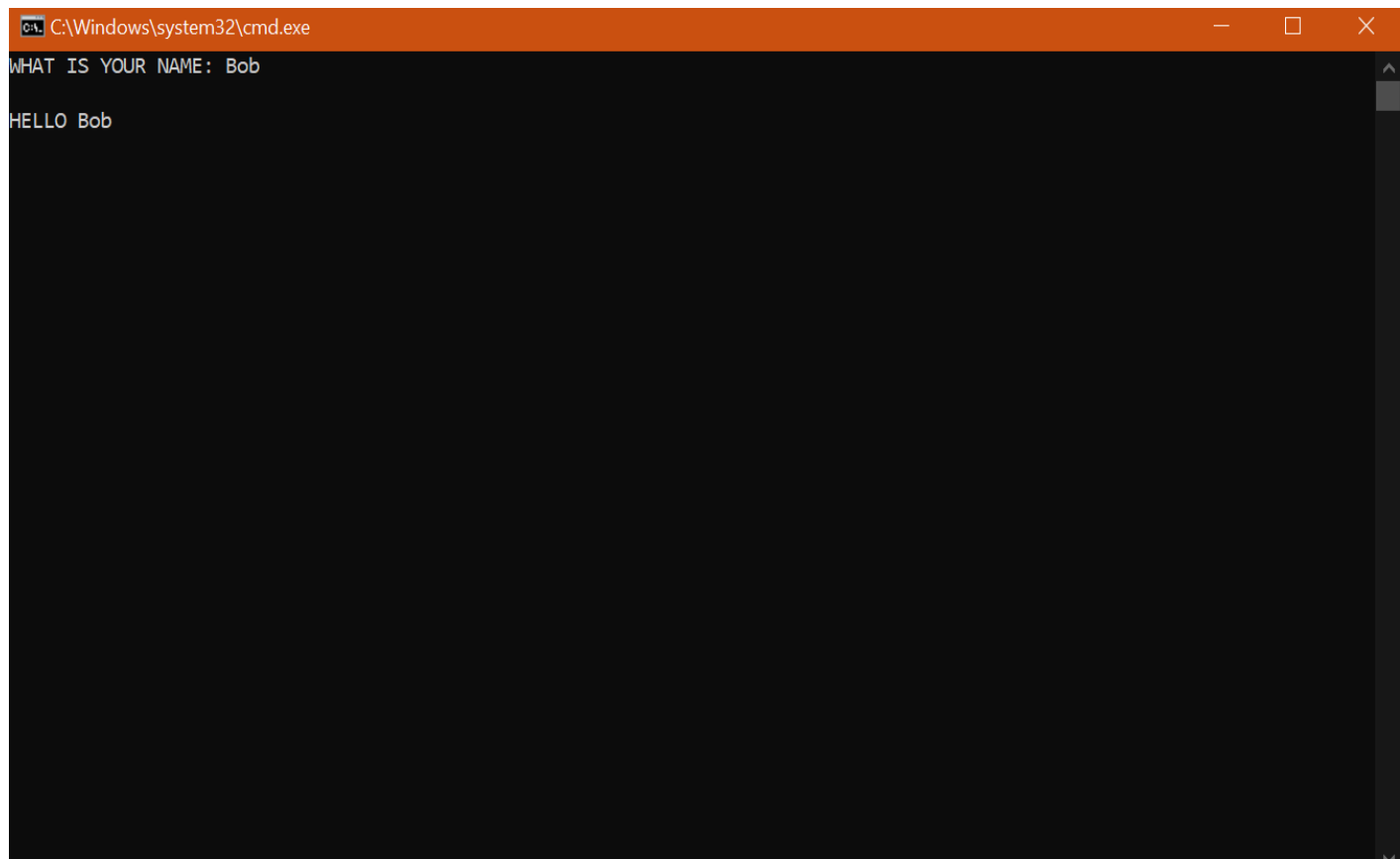
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;
using System.Threading.Tasks;

namespace Question3
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string name;
            Console.Write("WHAT IS YOUR NAME: ");
```

```
name = Console.ReadLine();
if(name == "Alice" || name == "Bob") {
    Console.WriteLine($"{nHELLO {name}");
    Console.ReadLine();
}

}

}
```



A screenshot of a Windows command prompt window. The title bar is orange and displays the file path "C:\Windows\system32\cmd.exe" along with standard window control buttons (minimize, maximize, close). The command prompt has a black background with white text. The first line of text is "WHAT IS YOUR NAME: Bob", which represents the user's input. The second line of text is "HELLO Bob", which is the output of the program. A vertical scrollbar is visible on the right side of the window.

Q4. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n

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

namespace Question4
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int num, sum=0;
            Console.Write("GIVE A NUMBER: ");
            num =Convert.ToInt32(Console.ReadLine());
            for(int i = 1; i <= num; i++)
            {
                sum += i;
            }
            Console.WriteLine($"TOTAL SUM FROM 1 TO {num} IS :
{sum}");
            Console.ReadLine();
        }
    }
}
```

```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question4\Question4\bin\Debug\Question4.exe
GIVE A NUMBER: 5
TOTAL SUM FROM 1 TO 5 IS : 15
```

Q5. Modify the previous program such that only multiples of three or five are considered in the sum, e.g. 3, 5, 6, 9, 10, 12, 15 for n=17.

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

namespace Question5
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int num, sum = 0;
            Console.Write("GIVE A NUMBER: ");
            num = Convert.ToInt32(Console.ReadLine());

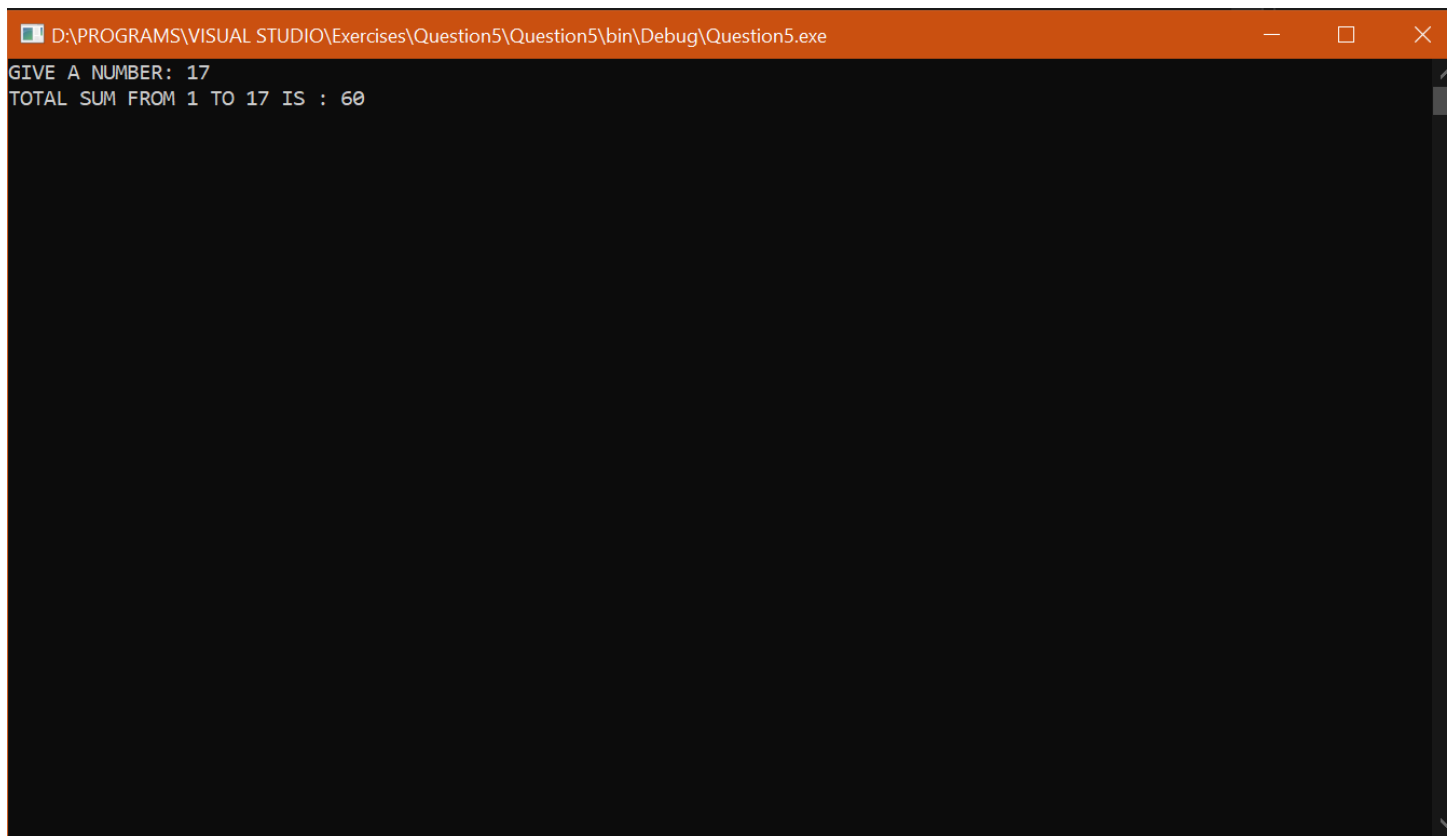
            for (int i = 1; i <= num; i++)
            {
                if (i % 3 == 0 || i % 5 == 0)
                {
                    sum += i;
                }
            }

            Console.WriteLine("TOTAL SUM FROM 1 TO " + num + " IS : " + sum);
        }
    }
}
```

```

        sum += i;
    }
}
Console.WriteLine($"TOTAL SUM FROM 1 TO {num} IS :
{sum}");
Console.ReadLine();
}
}
}

```



```

D:\PROGRAMS\VISUAL STUDIO\Exercises\Question5\Question5\bin\Debug\Question5.exe
GIVE A NUMBER: 17
TOTAL SUM FROM 1 TO 17 IS : 60

```

Q6. Write a program that asks the user for a number n and gives them the possibility to choose between computing the sum and computing the product of $1, \dots, n$.

```
using System;
```



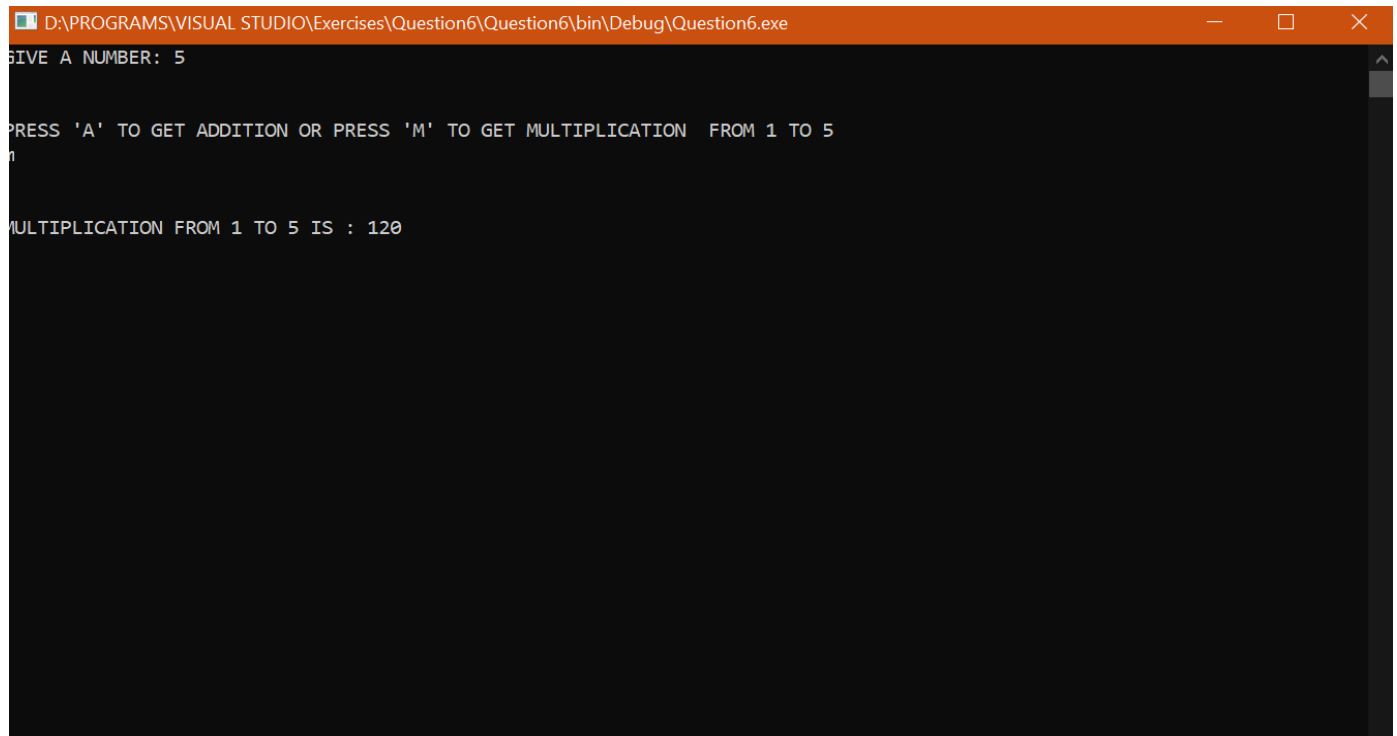
```

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

namespace Question6
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int num, sum = 0, mul = 1;
            char sel;
            Console.Write("GIVE A NUMBER: ");
            num = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine($"
PRESS 'A' TO GET ADDITION OR
PRESS 'M' TO GET MULTIPLICATION FROM 1 TO {num}");
            sel = (char) Console.Read();
            Console.ReadLine();
            if (sel == 'A' || sel == 'a')
            {
                for (int i = 1; i <= num; i++)
                {
                    sum += i;
                }
                Console.WriteLine($"TOTAL SUM FROM 1 TO {num} IS :
{sum}");
            }
            else if (sel == 'M' || sel == 'm')
            {
                for (int i = 1; i <= num; i++)
                {
                    mul *= i;
                }
                Console.WriteLine($"
MULTIPLICATION FROM 1 TO
{num} IS : {mul}");
            }

            Console.ReadLine();
        }
    }
}

```



```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question6\Question6\bin\Debug\Question6.exe
GIVE A NUMBER: 5
PRESS 'A' TO GET ADDITION OR PRESS 'M' TO GET MULTIPLICATION FROM 1 TO 5
MULTIPLICATION FROM 1 TO 5 IS : 120
```

Q7. Write a program that prints a multiplication table for numbers up to 12.

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

namespace Question7
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("TABLES UPTO 12: \n\n");
            for(int i = 1; i <= 12; i++) {
```

```

        Console.WriteLine($"\\nTABLE OF {i} UPTO 10: \\n");
        for (int j = 1; j <= 10; j++) {
            Console.WriteLine($"{i} x {j} = {i*j}");
        }
    }
    Console.ReadLine();
}
}
}

```

```

C:\Windows\system32\cmd.exe
TABLES UPTO 12:

TABLE OF 1 UPTO 10:
1 x 1 = 1
1 x 2 = 2
1 x 3 = 3
1 x 4 = 4
1 x 5 = 5
1 x 6 = 6
1 x 7 = 7
1 x 8 = 8
1 x 9 = 9
1 x 10 = 10

TABLE OF 2 UPTO 10:
2 x 1 = 2
2 x 2 = 4
2 x 3 = 6
2 x 4 = 8
2 x 5 = 10
2 x 6 = 12
2 x 7 = 14
2 x 8 = 16
2 x 9 = 18
2 x 10 = 20

TABLE OF 3 UPTO 10:
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24

```

Q8. Write a program that prints all prime numbers. (Note: if your programming language does not support arbitrary size numbers, printing all primes up to the largest number you can easily represent is fine too.)

```

using System;
using System.Collections.Generic;
using System.Linq;

```

```

using System.Text;
using System.Threading.Tasks;

namespace Question8
{
    internal class Program
    {
        static void Main(string[] args)
        {
            long num;
            Console.Write("GIVE A RANGE OF PRIME NUMBERS: ");
            num=Convert.ToInt64(Console.ReadLine());
            for(int i =2;i<num ;i++) {
                for (int j=2;j<i;j++)
                {
                    if (i % j == 0)
                    {
                        i++;
                        j = 2;
                    }

                }
                Console.Write($"{i}\t");
            }
            Console.ReadLine();
        }
    }
}

```

```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question8\Question8\bin\Debug\Question8.exe
GIVE A RANGE OF PRIME NUMBERS: 500
2  3    5    7    11   13   17   19   23   29   31   37   41   43   47
   53   59   61   67   71   73   79   83   89   97  101  103  107  109
  113  127  131  137  139  149  151  157  163  167  173  179  181  191
  193  197  199  211  223  227  229  233  239  241  251  257  263  269
  271  277  281  283  293  307  311  313  317  331  337  347  349  353
  359  367  373  379  383  389  397  401  409  419  421  431  433  439
  443  449  457  461  463  467  479  487  491  499
```

Q9. Write a guessing game where the user has to guess a secret number. After every guess the program tells the user whether their number was too large or too small. At the end the number of tries needed should be printed. It counts only as one try if they input the same number multiple times consecutively.

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

namespace Question9
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int num, guess=0;
            Random rnd = new Random();
            num=rnd.Next(1,500);
        }
    }
}
```

```

        Console.WriteLine("-----
-TELL US HOW SMART YOU ARE-----");
        Console.Write("\nGuess a number between 1 to 500: ");
        int count = 0, prev;

        do
        {
            prev = guess;
            guess = Convert.ToInt32(Console.ReadLine());

            if (guess < num)
            {
                Console.Write($"Try Some Bigger Number: ");
            }
            else if (guess > num)
            {
                Console.Write("\nTry Some Smaller Number: ");
            }
            if (prev != guess) {
                count++;
            }

        } while (guess != num);
        Console.WriteLine($"CONGRATULATIONS YOU GUESSED
THE CORRECT NUMBER IN {count} TRIES");
        Console.ReadLine();
    }
}

```

```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question9\Question9\bin\Debug\Question9.exe
-----TELL US HOW SMART YOU ARE-----
Guess a number between 1 to 500: 40
Try Some Bigger Number: 100
Try Some Bigger Number: 200
Try Some Bigger Number: 400
Try Some Bigger Number: 450
Try Some Smaller Number: 440
Try Some Bigger Number: 445
Try Some Bigger Number: 449
CONGRATULATIONS YOU GUESSED THE CORRECT NUMBER IN 8 TRIES
```

Q10. Write a program that prints the next 20 leap years.

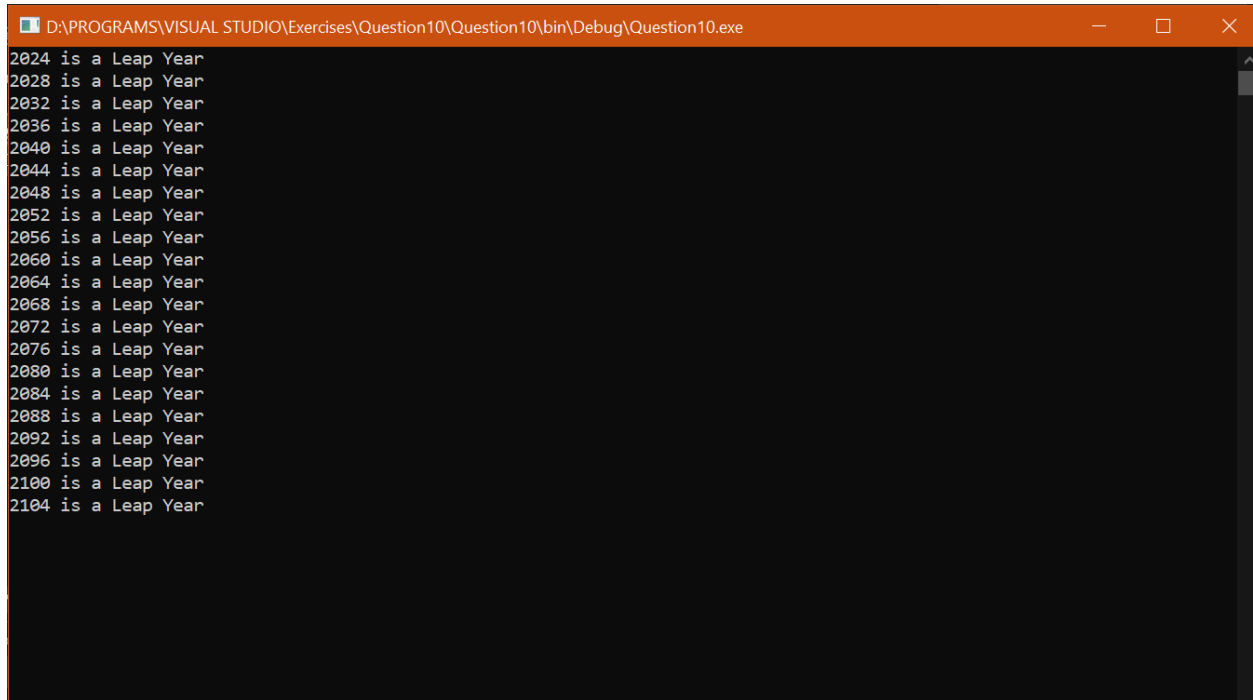
```
using System;

namespace Question10
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int presentYear=2023,count=0;
            while (count<=20) {
                if (presentYear%4==0) {
                    Console.WriteLine($"{presentYear } is a Leap
Year");
                    count++;
                }
            }
        }
    }
}
```

```

        presentYear++;
    }
    Console.ReadLine();
}
}
}

```



```

D:\PROGRAMS\VISUAL STUDIO\Exercises\Question10\Question10\bin\Debug\Question10.exe
2024 is a Leap Year
2028 is a Leap Year
2032 is a Leap Year
2036 is a Leap Year
2040 is a Leap Year
2044 is a Leap Year
2048 is a Leap Year
2052 is a Leap Year
2056 is a Leap Year
2060 is a Leap Year
2064 is a Leap Year
2068 is a Leap Year
2072 is a Leap Year
2076 is a Leap Year
2080 is a Leap Year
2084 is a Leap Year
2088 is a Leap Year
2092 is a Leap Year
2096 is a Leap Year
2100 is a Leap Year
2104 is a Leap Year

```

Q11. Write a function that returns the largest element in a list.

```

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

namespace Question11
{
    internal class Program
    {
        public static int maximumEle(int[] arr)
        {
            return arr.Max();
        }
    }
}

```



```

static void Main(string[] args)
{
    int[] arr = new int[10];
    Console.WriteLine("GIVE 10 NUMBERS:");
    for(int i =0; i < arr.Length; i++) {
        arr[i]=Convert.ToInt32(Console.ReadLine());
    }
    Console.WriteLine("Largest Number is
"+maximumEle(arr));
    Console.ReadLine();
}
}
}

```

```

D:\PROGRAMS\VISUAL STUDIO\Exercises\Question11\Question11\bin\Debug\Question11.exe
GIVE 10 NUMBERS:
1
2
3
4
5
34
5
6
7
8
Largest Number is 34

```

Q12. Write function that reverses a list, preferably in place.

```

using System;
using System.Collections.Generic;

```

```

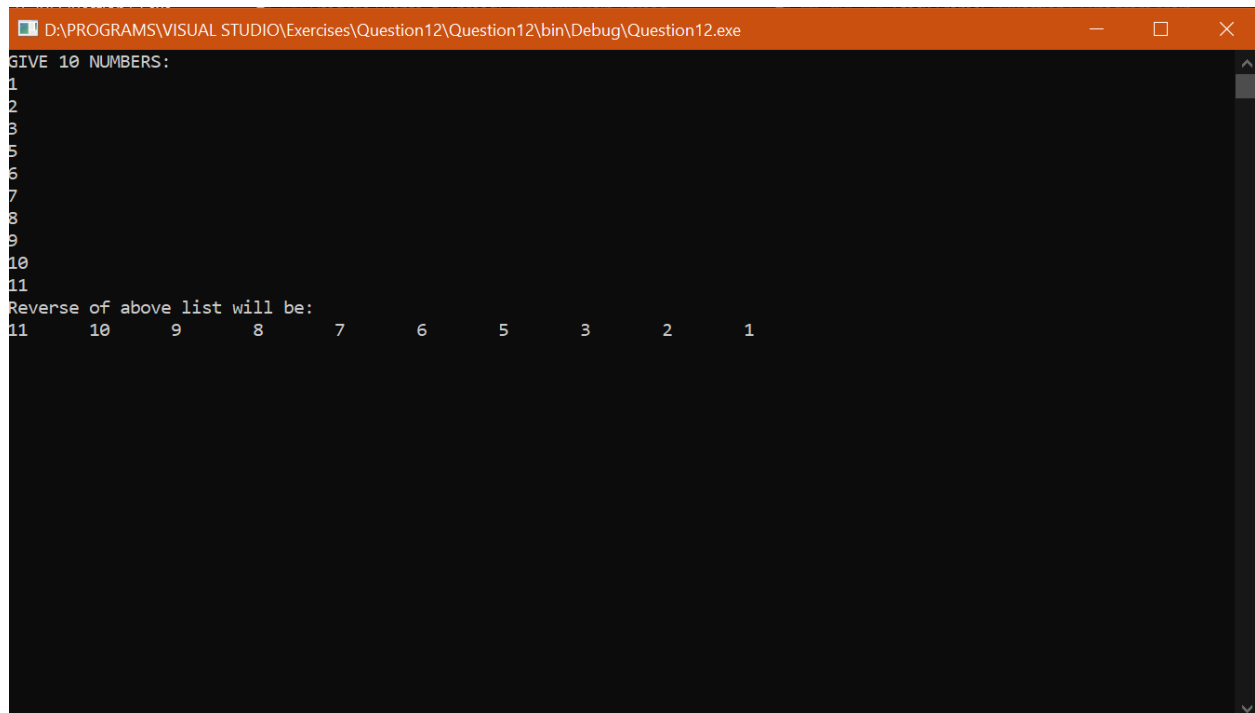
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Question12
{
    internal class Program
    {
        public static int[] reverse(int[] arr, int[] lst)
        {
            int j = arr.Length - 1;
            for (int i = 0; i < arr.Length; i++)
            {
                lst[i] = arr[j];
                j--;
            }
            return lst;
        }

        static void Main(string[] args)
        {
            int[] lst = new int[10];
            int[] arr = new int[10];
            Console.WriteLine("GIVE 10 NUMBERS:");
            for (int i = 0; i < arr.Length; i++)
            {
                arr[i] = Convert.ToInt32(Console.ReadLine());
            }

            Console.WriteLine("Reverse of above list will be: ");
            lst = reverse(arr, lst);
            foreach(int i in lst)
            {
                Console.WriteLine(i);
            }
            Console.ReadLine();
        }
    }
}

```



```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question12\Question12\bin\Debug\Question12.exe
GIVE 10 NUMBERS:
1
2
3
5
6
7
8
9
10
11
Reverse of above list will be:
11      10      9      8      7      6      5      3      2      1
```

Q13. Write a function that checks whether an element occurs in a list.

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

namespace Question13
{
    internal class Program
    {
        public static bool checkEle(int[] list, int element)
        {
            return list.Contains(element);
        }

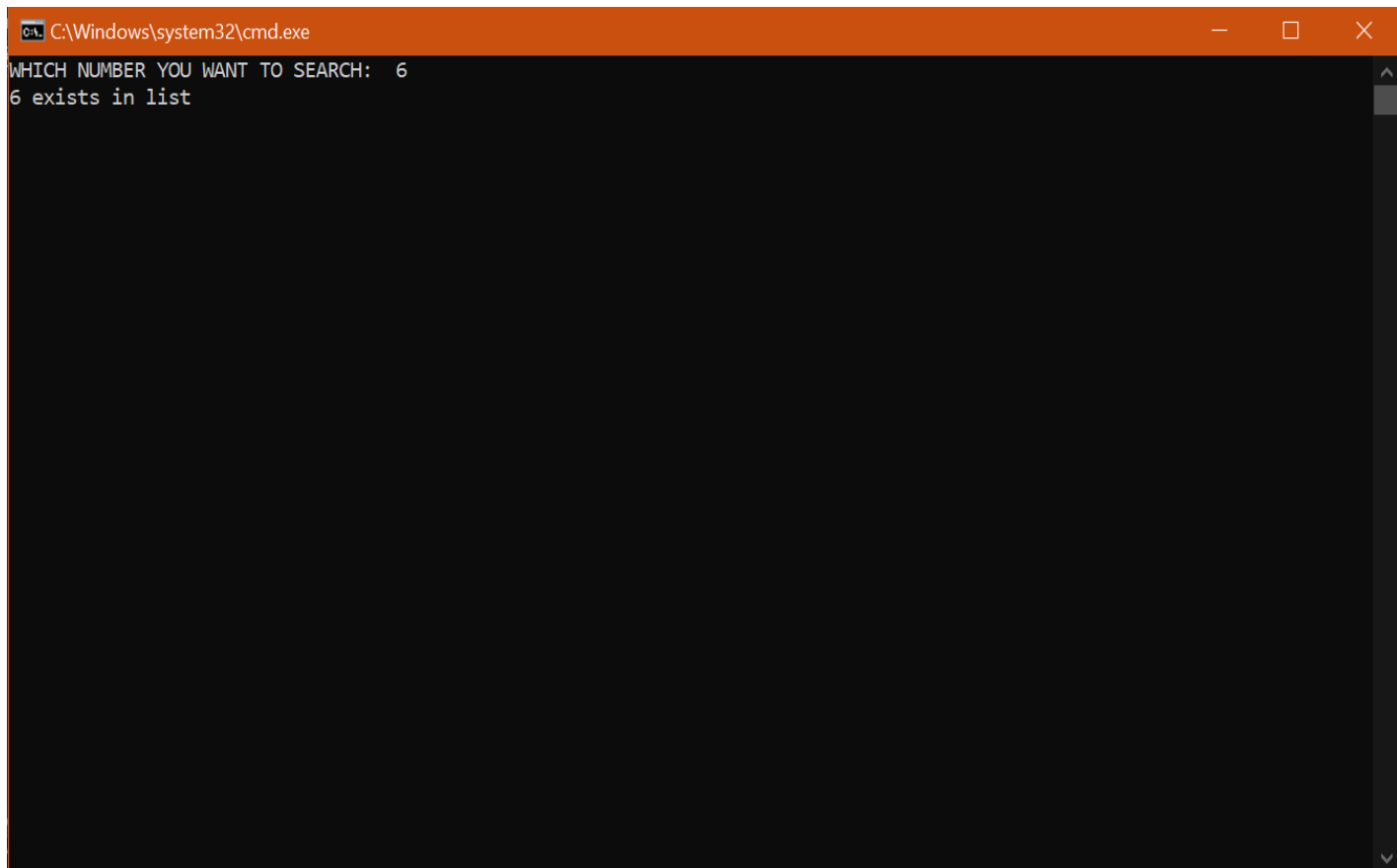
        static void Main(string[] args)
        {
            int[] arr = {1,2,3,4,5,6 };
        }
    }
}
```

```
int num;
Console.Write("WHICH NUMBER YOU WANT TO SEARCH: ");
num=Convert.ToInt32(Console.ReadLine());

if(checkEle(arr,num)) {
    Console.WriteLine($"{num} exists in list");
}
else
{
    Console.WriteLine("IT DOESN'T EXIST");
}

Console.ReadLine();

    }
}
}
```



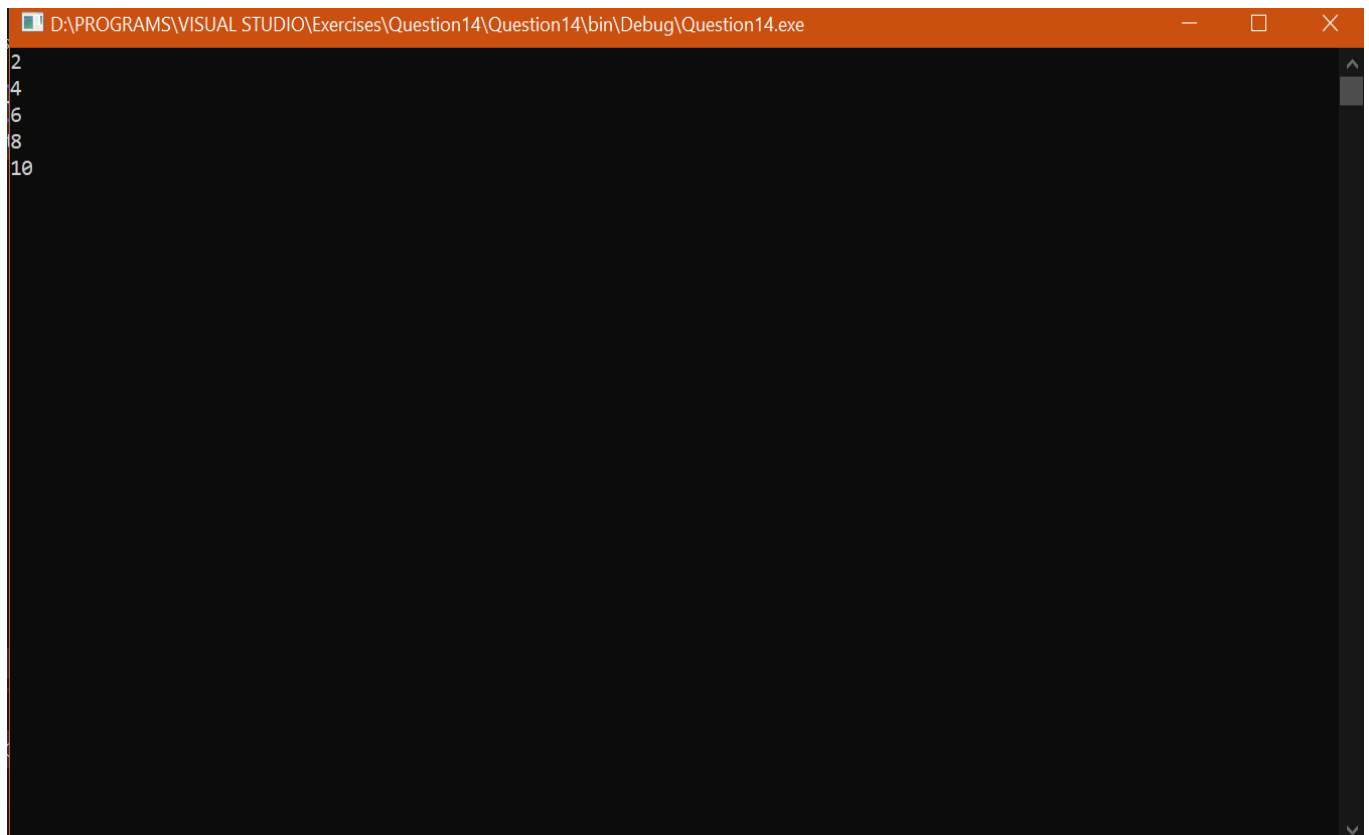
A screenshot of a Windows command prompt window. The title bar is orange and displays the path "C:\Windows\system32\cmd.exe" along with standard window controls. The command prompt has a black background with white text. It shows the prompt "WHICH NUMBER YOU WANT TO SEARCH: " followed by the user input "6". Below this, the program's output "6 exists in list" is displayed. The rest of the window is empty.

```
C:\Windows\system32\cmd.exe
WHICH NUMBER YOU WANT TO SEARCH: 6
6 exists in list
```

Q14. Write a function that returns the elements on odd positions in a list

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

namespace Question14
{
    internal class Program
    {
        public static int[] GetOddPositions(int[] lst)
        {
            return lst.Where((ind, val) => val % 2 ==
1).ToArray();
        }
        static void Main(string[] args)
        {
            int[] arr = {1,2,3,4,5,6,7,8,9,10};
            int[] oddArr = GetOddPositions(arr);
            for(int i = 0; i < oddArr.Length; i++)
            {
                Console.WriteLine(oddArr[i]);
            }
            Console.ReadLine();
        }
    }
}
```



Q15. Write a function that computes the running total of a list.

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

namespace Question15
{
    internal class Program
    {
        public static int runningTotal(int[] numbers)
        {
            int sum=0;
            for (int n = 0; n < numbers.Length; n++)
            {
                sum += numbers[n];
            }
            return sum;
        }
    }
}
```

```

    }
    static void Main(string[] args)
    {
        int[] numbers = new int[10];
        int res ;

        Console.WriteLine("GIVE 10 NUMBERS:");
        for (int i = 0; i < numbers.Length; i++)
        {
            numbers[i] = Convert.ToInt32(Console.ReadLine());
        }
        res = runningTotal(numbers);

        Console.WriteLine($"THE RUNNING TOTAL IS: {res}");

        Console.ReadLine();
    }
}

```

```

C:\Windows\system32\cmd.exe
GIVE 10 NUMBERS:
1
2
3
4
5
6
7
8
9
10
THE RUNNING TOTAL IS: 55

```

Q16. Write a function that tests whether a string is a palindrome.

```

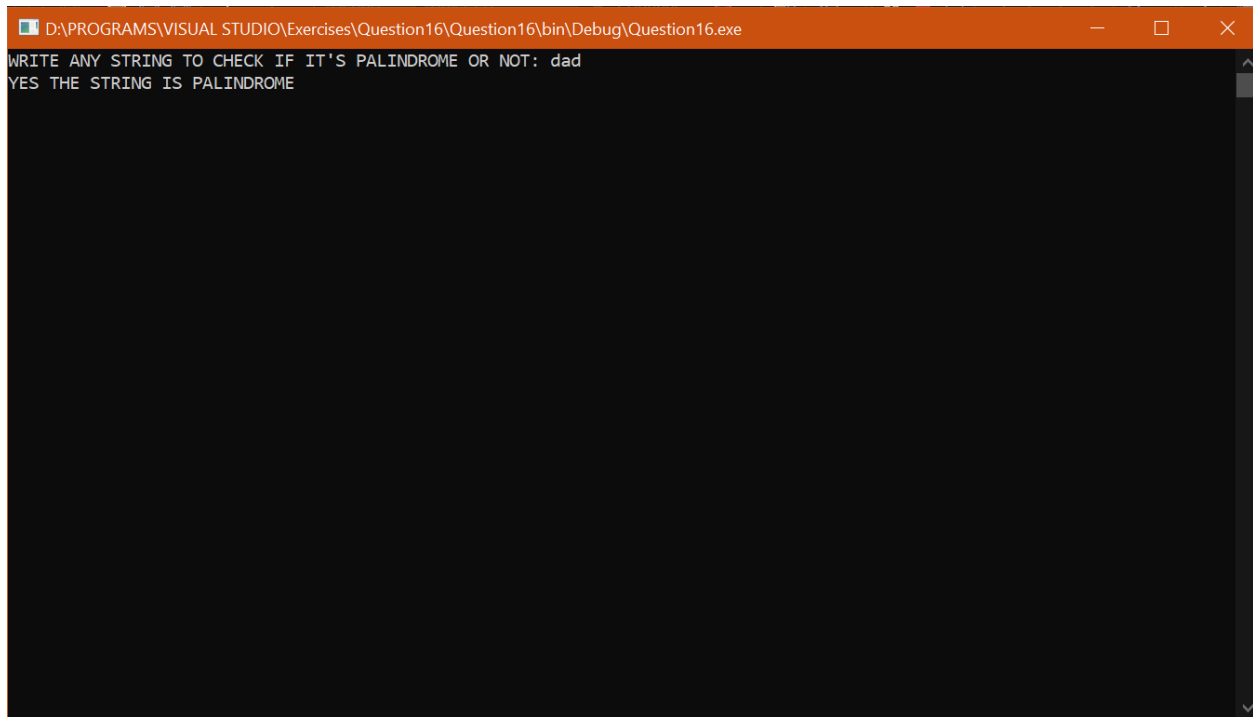
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

namespace Question16
{
    internal class Program
    {
        public static string checkPalindrome(char[] str)
        {
            string rev = new string(str.Reverse().ToArray());
            string toStr = new string(str);
            if (rev == toStr)
            {
                return "YES THE STRING IS PALINDROME";
            }
            else
            {
                return "NO THE STRING IS NOT PALINDROME";
            }
        }
        static void Main(string[] args)
        {
            char[] str;
            Console.Write("WRITE ANY STRING TO CHECK IF IT'S
PALINDROME OR NOT: ");
            str = Console.ReadLine().ToCharArray();
            Console.WriteLine(checkPalindrome(str));
            Console.ReadLine();
        }
    }
}

```

Q17. Write three functions that compute the sum of the numbers in a list: using a for-loop, a while-loop and recursion. (Subject to availability of these constructs in your language of choice.)

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

namespace Question17
{
    internal class Program
    {
        public static int sumFor(int[] arrNum)
        {
            int sum = 0;
            for (int i = 0; i < arrNum.Length; i++)
            {
                sum += arrNum[i];
            }
            return sum;
        }
    }
}
```

```

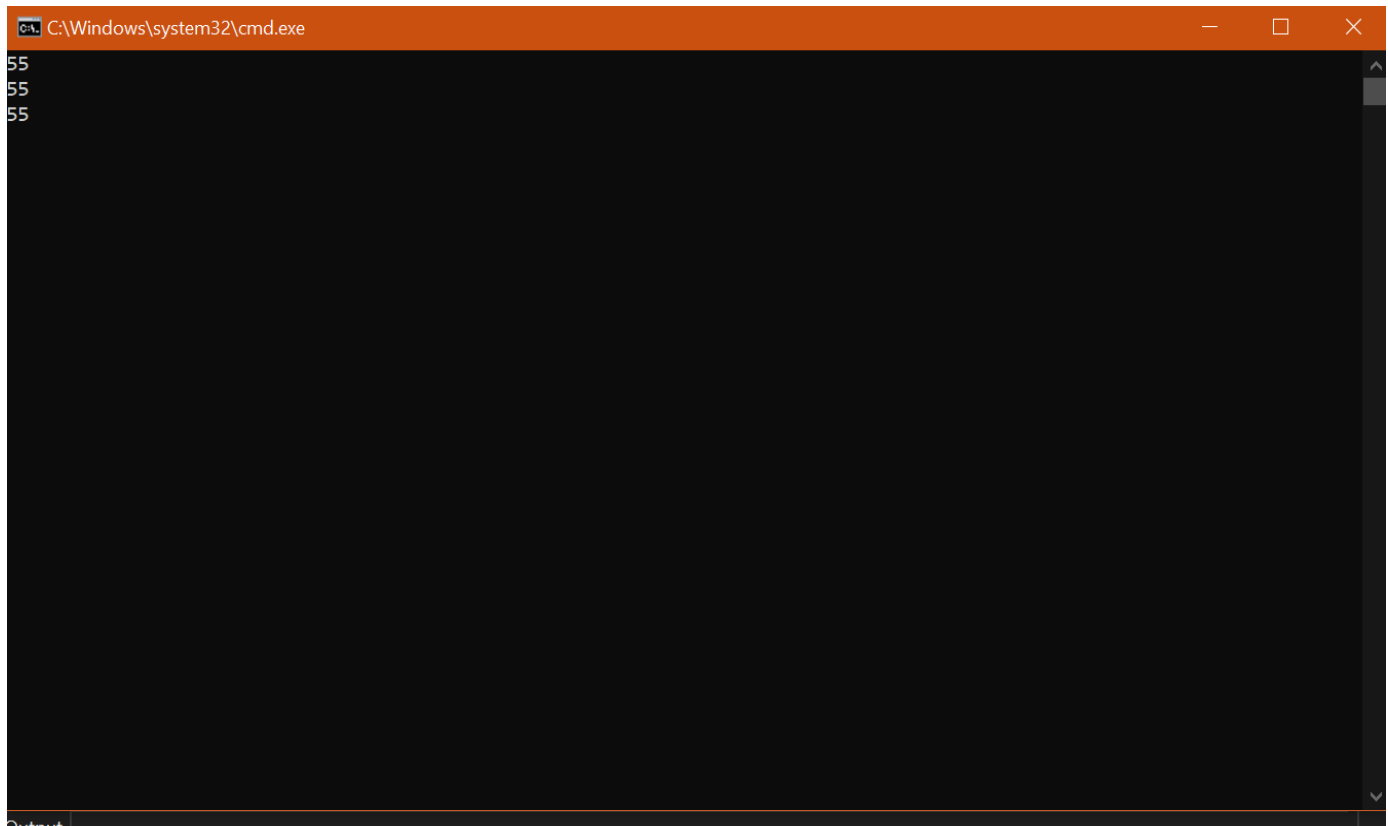
public static int sumWhile(int[] arrNum)
{
    int sum = 0;
    int i = 0;
    while (i < arrNum.Length)
    {
        sum += arrNum[i];
        i++;
    }
    return sum;
}

public static int sumRec(int[] arrNum, int ind)
{
    if (ind == arrNum.Length)
    {
        return 0;
    }
    else
    {
        return arrNum[ind] + sumRec(arrNum, ind + 1);
    }
}

public static void Main()
{
    int[] arrNum = { 1, 2, 3, 4, 5,6,7,8,9,10 };

    Console.WriteLine(sumFor(arrNum));
    Console.WriteLine(sumWhile(arrNum));
    Console.WriteLine(sumRec(arrNum, 0));
    Console.ReadLine();
}
}
}

```



```
C:\Windows\system32\cmd.exe
55
55
55
```

Q18. Write a function `on_all` that applies a function to every element of a list. Use it to print the first twenty perfect squares. The perfect squares can be found by multiplying each natural number with itself. The first few perfect squares are $1*1=1$, $2*2=4$, $3*3=9$, $4*4=16$. Twelve for example is not a perfect square because there is no natural number m so that $m*m=12$. (This question is tricky if your programming language makes it difficult to pass functions as arguments.)

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

namespace Question18
{
    internal class Program
    {
```

```

delegate int fun(int x);
// Function that applies a function to every element of a
list
static List<int> On_All(List<int> list, fun func)
{
    List<int> result = new List<int>();
    foreach (int item in list)
    {
        result.Add(func(item));
    }
    return result;
}

// Function that prints the first twenty perfect squares
static int PerfectSquares(int n) => n * n;

static void Main()
{
    // Call the function and print the first twenty
perfect squares
    List<int> numbers= new List<int>(20);
    for (int i=1;i <= 20;i++) {
        numbers.Add(i);
    }
    List<int> res = On_All(numbers,PerfectSquares);
    Console.WriteLine("PRINTING PERFECT SQUARES OF FIRST
20 NATURAL NUMBERS: ");
    Console.Write("[ ");
    foreach (int i in res)
    {
        Console.Write(i+" ");
    }
    Console.WriteLine(" ]");
    Console.ReadLine();
}
}
}

```

```
D:\PROGRAMS\VISUAL STUDIO\Exercises\Question18\Question18\bin\Debug\Question18.exe
PRINTING PERFECT SQUARES OF FIRST 20 NATURAL NUMBERS:
[ 1 4 9 16 25 36 49 64 81 100 121 144 169 196 225 256 289 324 361 400 ]
```

Q19. Write a function that computes the list of the first 100 Fibonacci numbers. The first two Fibonacci numbers are 1 and 1. The $n+1$ -st Fibonacci number can be computed by adding the n -th and the $n-1$ -th Fibonacci number. The first few are therefore 1, 1, $1+1=2$, $1+2=3$, $2+3=5$, $3+5=8$.

LOGIC FILTERATION:

$i=0$

$(i==1 \text{ or } i==0) \Rightarrow$ add 1 into array

$\text{arr}[i-1] + \text{arr}[i-2]$ add into array

i.e; $i=2$

$\text{arr}[2-1] \Rightarrow \text{arr}[1]=1$

$\text{arr}[2-2] \Rightarrow \text{arr}[0]=1$

$1+1=2 \Rightarrow$ when $i=2$ fibonacci will produce 2 but,

$i=3$

$\text{arr}[3-1] \Rightarrow \text{arr}[2]=2$, $\text{arr}[3-2]=\text{arr}[1]=1$

2+1=3> when i =3 fibonacci will produce 3 but when it becomes 4

arr[4-1] => arr[3] = 3 ,

arr[4-2]=>arr[2]=2

3+2=5>= similary it will be producing sequence by adding up two previous numbers.

CODE:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Threading.Tasks;
using static System.Net.Mime.MediaTypeNames;

namespace Question19
{
    internal class Program
    {
        static void FibonacciSeq(int n) {
            int i = 0;
            List<int> arrRes = new List<int>();
            while(true)
            {

                if (i == 0 || i == 1)
                {
                    arrRes.Add(1);
                }
                else
                {
                    arrRes.Add(arrRes[i - 1] + arrRes[i - 2]);
                    if (arrRes[i] > n)break;
                }
                Console.Write(arrRes[i] + " ");

                i++;
            }
            Console.WriteLine("");
        }
        static void Main(string[] args)
        {
```

```

        Console.WriteLine("GIVE US THE RANGE OF FIBONACCI
SEQUENCE: ");
        int range=Convert.ToInt32(Console.ReadLine());
        FibonacciSeq(range);
        Console.ReadLine();
    }
}
}

```

```

C:\Windows\system32\cmd.exe
GIVE US THE RANGE OF FIBONACCI SEQUENCE: 100
1 1 2 3 5 8 13 21 34 55 89

```

Q20. Write a function that takes a number and returns a list of its digits. So for 2342 it should return [2,3,4,2].

LOGIC FILTERATION:

```

Num=2342 >=convert into string str="2342"
arr[0]=str[0]=2
.
.
.
arr[n-1]=str[n-1]=2

```

```
arr=[2,3,4,2]
```

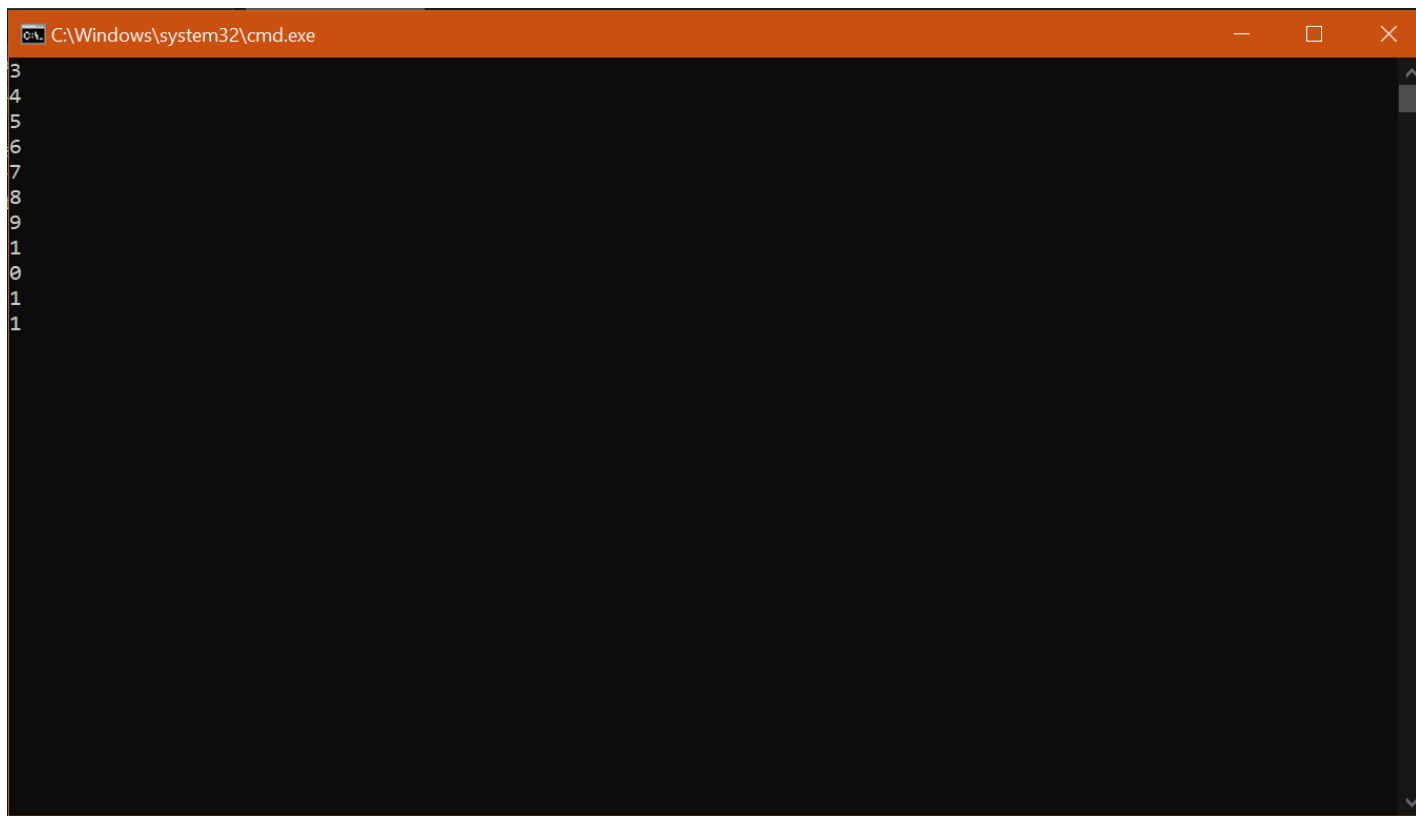
CODE:

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

namespace Question20
{
    internal class Program
    {
        static List<int> splitter(long num) {
            string str = num.ToString();

            List<int> digits = new List<int>();

            for(int s = 0; s < str.Length; s++)
            {
                digits.Add(Convert.ToInt32(str[s].ToString()));
            }
            return digits;
        }
        static void Main(string[] args)
        {
            List<int> digits= splitter(34567891011);
            foreach (int i in digits) {
                Console.WriteLine(i);
            }
            Console.ReadLine();
        }
    }
}
```

21. Write a function that takes a list of strings and prints them, one per line, in a rectangular frame. For example the list ["Hello", "World", "in", "a", "frame"] gets printed as:

```
*****
* Hello *
* World *
* in    *
* a     *
* frame *
*****
```

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

```

namespace Question21
{
    internal class Program
    {
        public static void PrintPattern(List<string> lst,int
maxLen)
        {

            int k = 0;
            for(int i = 0; i < (lst.Count()+2); i++)
            {
                Console.Write("*");
                for (int j=0;j<maxLen+2;j++)
                {

                    if (i != 0 && i != lst.Count() + 1)
                    {

                        if (k < lst.Count())
                        {
                            int r = maxLen - lst[k].Length;

                            Console.Write($" {lst[k]} ");
                            for (int l = 0; l < r; l++)
                            {
                                Console.Write(" ");
                            }
                            k++;
                        }
                        break;
                    }
                    else {
                        Console.Write("*");
                    }

                }

                Console.WriteLine("*");
            }

        }

        static void Main(string[] args)
        {

            string str ;

```

```

        Console.WriteLine("TYPE A STRING: ");
        str=Console.ReadLine();
        List<string> strArr = str.Split(' ').ToList();
        // find the length of the longest string
        int maxLen = 0;
        foreach (string s in strArr)
        {
            if (s.Length > maxLen)
            {
                maxLen = s.Length;
            }
        }
        PrintPattern(strArr,maxLen);
        Console.Read();
    }
}

```

```

C:\Windows\system32\cmd.exe
TYPE A STRING: Hello My Name is Uzair Ahmed
*****
* Hello *
* My    *
* Name  *
* is    *
* Uzair *
* Ahmed *
*****

```

Q22. Given two strings, write a program that efficiently finds the longest common subsequence.

:

```

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

namespace Question20
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string str1 = "zxabcdezy";
            string str2 = "yzabcdez";
            int[,] LCS = LcsTable(str1, str2);
            string lcs = FindLCS(str1, str2, LCS);
            Console.WriteLine($"
Longest Common Subsequence of
{str1} and {str2} is : {lcs}");
            Console.Read();
        }

        static int[,] LcsTable(string str1, string str2)
        {
            int[,] LCS = new int[str1.Length + 1, str2.Length + 1];

            for (int i = 0; i <= str1.Length; i++)
            {
                for (int j = 0; j <= str2.Length; j++)
                {
                    if (i == 0 || j == 0)
                    {
                        LCS[i, j] = 0;
                    }
                    else if (str1[i - 1] == str2[j - 1])
                    {
                        LCS[i, j] = LCS[i - 1, j - 1] + 1;
                    }
                    else
                    {
                        LCS[i, j] = Math.Max(LCS[i - 1, j], LCS[i, j -
1]);
                    }
                }
            }
        }
    }
}

```

```

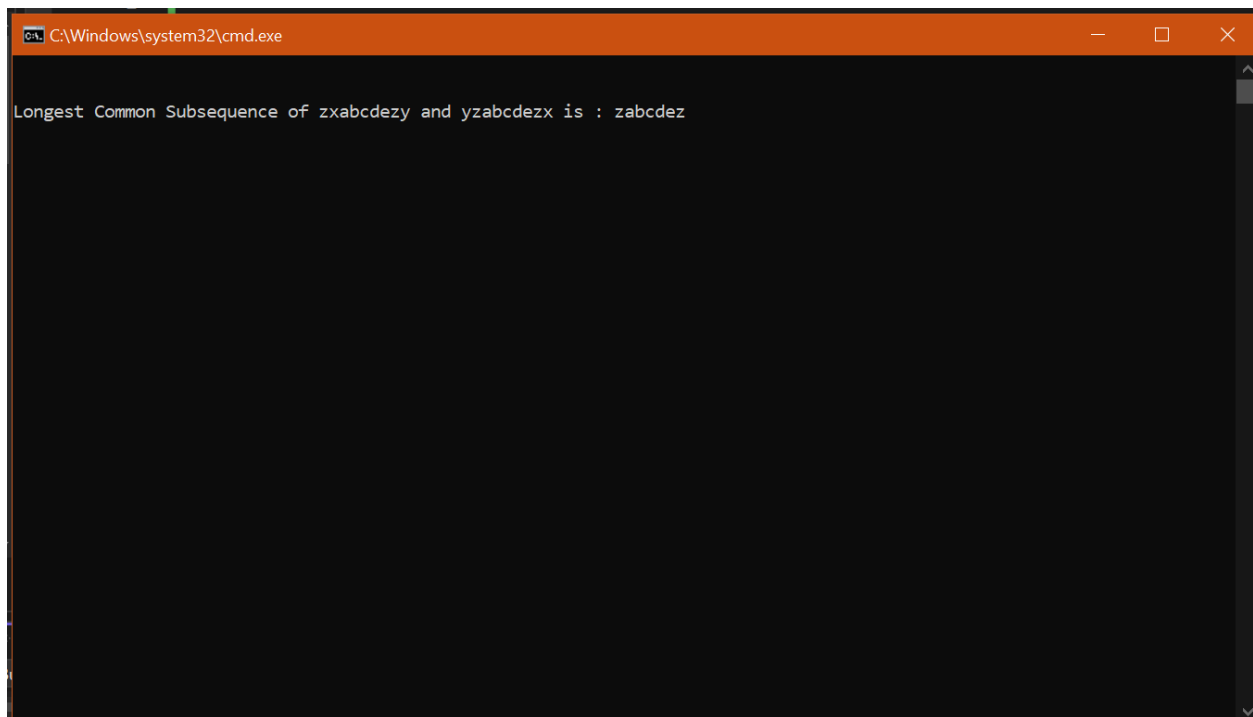
        return LCS;
    }

static string FindLCS(string str1, string str2, int[,] LCS)
{
    string finalLcs = "";
    int row = str1.Length;
    int col = str2.Length;

    while (row > 0 && col > 0)
    {
        if (str1[row - 1] == str2[col - 1])
        {
            finalLcs = str1[row - 1] + finalLcs;
            row--;
            col--;
        }
        else if (LCS[row - 1, col] > LCS[row, col - 1])
        {
            row--;
        }
        else
        {
            col--;
        }
    }

    return finalLcs;
}
}

```

A screenshot of a Windows command prompt window. The title bar is orange and shows the path 'C:\Windows\system32\cmd.exe'. The command prompt has a black background with white text. The text displayed is 'Longest Common Subsequence of zxabcdezy and yzabcdez is : zabcdez'.

```
C:\Windows\system32\cmd.exe

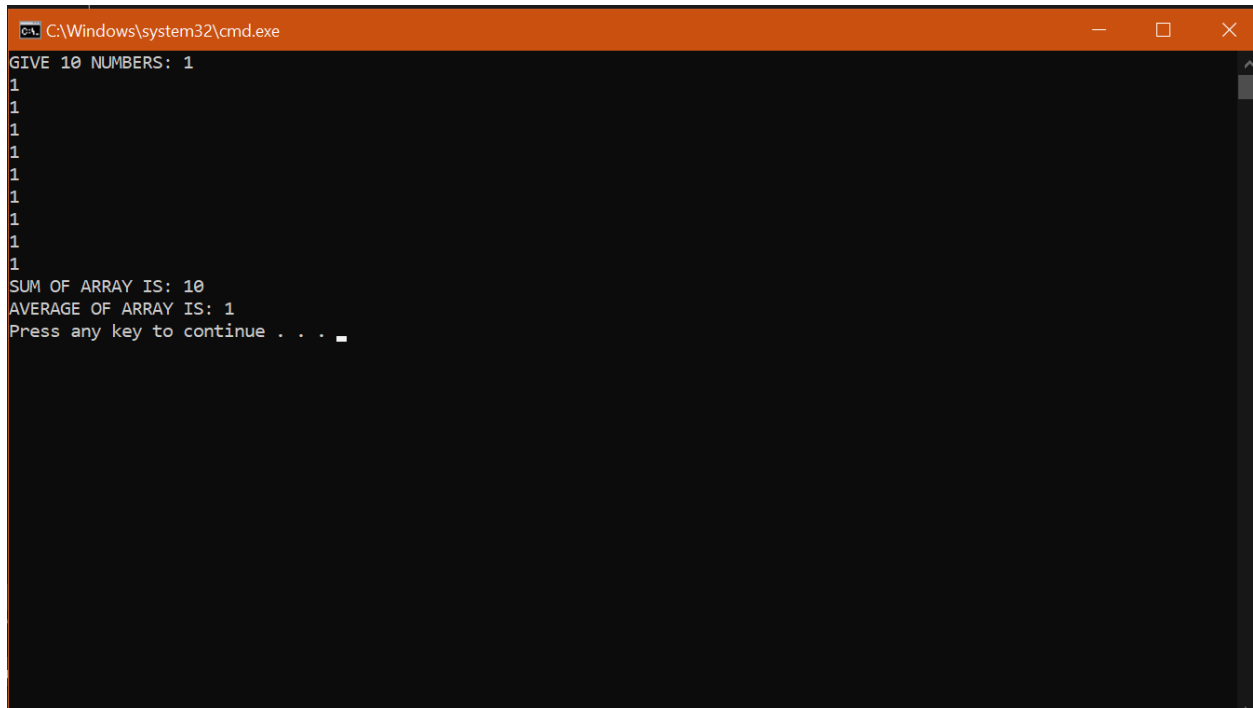
Longest Common Subsequence of zxabcdezy and yzabcdez is : zabcdez
```

Q23. Write a program in C to read 10 numbers from the keyboard and find their sum and average.

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

namespace Question23
{
    internal class Program
    {
        static void Main(string[] args)
        {
            List<int> nums = new List<int>(10);
            Console.Write("GIVE 10 NUMBERS: ");
            for(int i=0;i<10;i++) {
                nums.Add(Convert.ToInt32(Console.ReadLine()));
            }
            int sumOfArr=nums.Sum();
            double avgOfArr=nums.Average();
            Console.WriteLine($"SUM OF ARRAY IS: {sumOfArr}");
            Console.WriteLine($"AVERAGE OF ARRAY IS: {avgOfArr}");
        }
    }
}
```

```
}  
}
```



A screenshot of a Windows command prompt window with an orange title bar. The title bar text is "C:\Windows\system32\cmd.exe". The window contains the following text:
GIVE 10 NUMBERS: 1
1
1
1
1
1
1
1
1
1
1
1
1
1
SUM OF ARRAY IS: 10
AVERAGE OF ARRAY IS: 1
Press any key to continue . . .

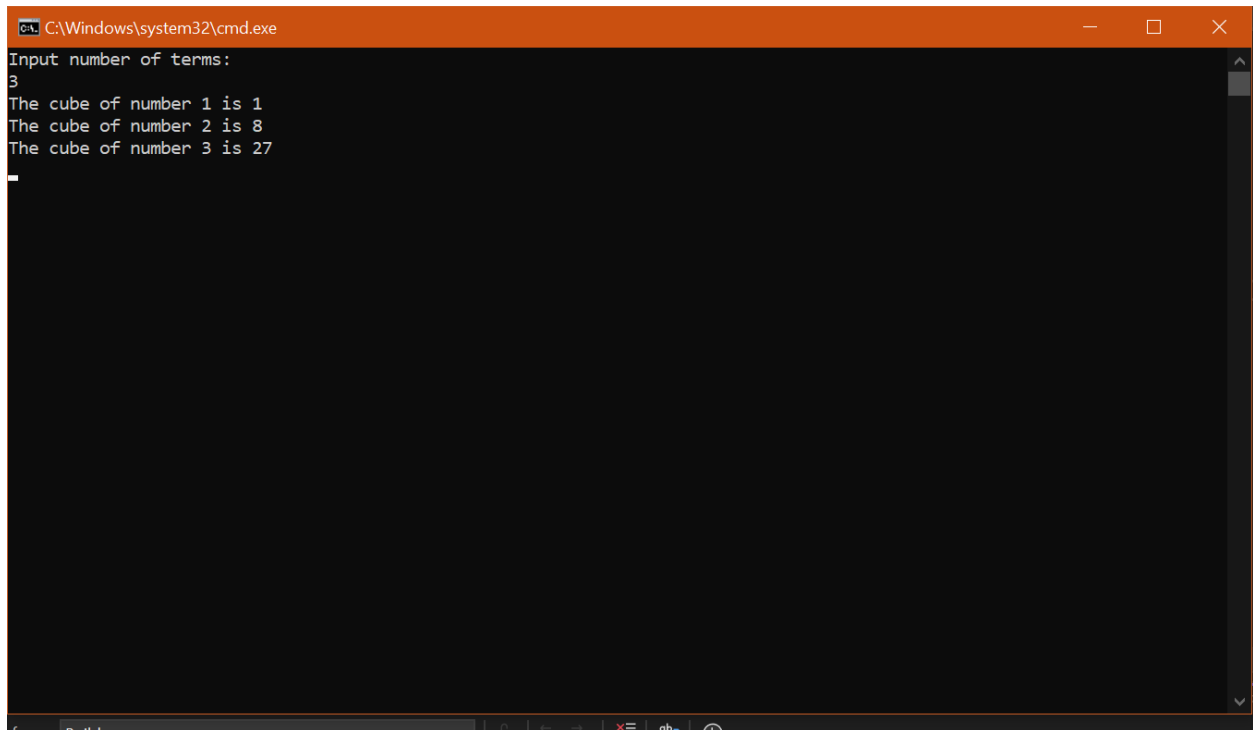
Q24. Write a program in C to display the cube of the number up to an integer

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
  
namespace Question24  
{  
    internal class Program  
    {  
        static void Main(string[] args)  
        {
```

```

        Console.WriteLine("Input number of terms: ");
        int n = Convert.ToInt32(Console.ReadLine());
        int cube;
        for (int i = 1; i <= n; i++)
        {
            cube = i * i * i;
            Console.WriteLine($"The cube of number {i} is
{cube}");
        }
        Console.ReadLine();
    }
}
}

```



```

C:\Windows\system32\cmd.exe
Input number of terms:
3
The cube of number 1 is 1
The cube of number 2 is 8
The cube of number 3 is 27

```

25. Write a program in C to display the multiplier table vertically from 1 to n.

```

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

```



```

namespace Question25
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Input upto the table number
starting from 1: ");
            int n = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Multiplication table from 1 to
{0}", n);
            for (int i = 1; i <= 10; i++)
            {
                for (int j = 1; j <= n; j++)
                {
                    Console.Write($"{j}x{i} = {i*j}");
                }
                Console.WriteLine();
            }
            Console.ReadLine();
        }
    }
}

```

```

C:\Windows\system32\cmd.exe
Input upto the table number starting from 1:
4
Multiplication table from 1 to 4
1x1 = 1,    2x1 = 2,    3x1 = 3,    4x1 = 4,
1x2 = 2,    2x2 = 4,    3x2 = 6,    4x2 = 8,
1x3 = 3,    2x3 = 6,    3x3 = 9,    4x3 = 12,
1x4 = 4,    2x4 = 8,    3x4 = 12,   4x4 = 16,
1x5 = 5,    2x5 = 10,   3x5 = 15,   4x5 = 20,
1x6 = 6,    2x6 = 12,   3x6 = 18,   4x6 = 24,
1x7 = 7,    2x7 = 14,   3x7 = 21,   4x7 = 28,
1x8 = 8,    2x8 = 16,   3x8 = 24,   4x8 = 32,
1x9 = 9,    2x9 = 18,   3x9 = 27,   4x9 = 36,
1x10 = 10,  2x10 = 20,  3x10 = 30,  4x10 = 40,

```

Q26. Write a C program to check whether a given number is an Armstrong number or not.

LOGIC FILTERATION:

num=153 =>will separate all digits

a=153%10=3

b=(153/10)%10=5

c=153/100=1,

Now every digit will get power upto the total number of digits

e=a^3=3^3=27

f=b^3=5^3=125

g=c^3=1^3=1

After giving power to them they will be added to each other.

total=e+f+g=>27+125+1

total==num =>NUM IS AN ARMSTRONG NUMBER.

CODE:

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

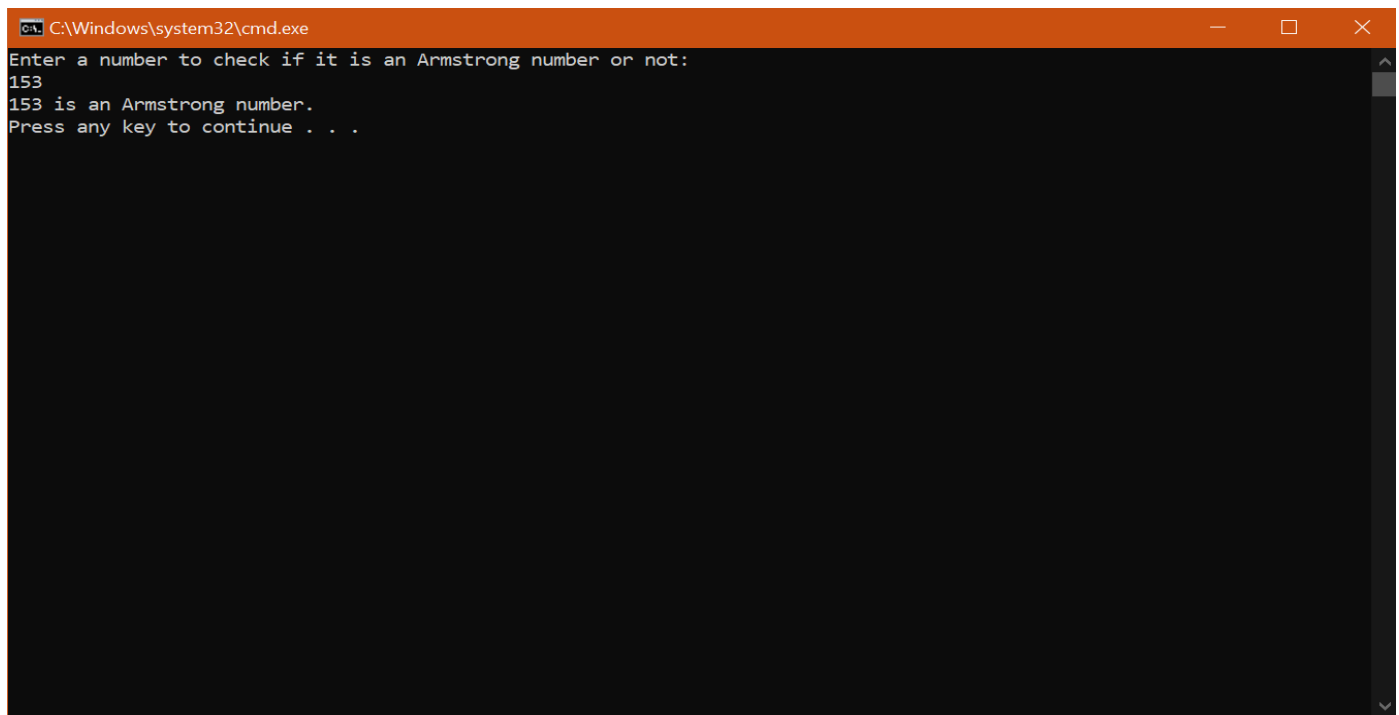
namespace Question26
{
    internal class Program
    {
        static void Main(string[] args)
        {
```

```

        Console.WriteLine("Enter a number to check if it is an
Armstrong number or not: ");
        int n = int.Parse(Console.ReadLine());
        int temp = n;
        int sum = 0;
        int digits = countDigits(n);
        while (n > 0)
        {
            int rem = n % 10;
            sum += (int)Math.Pow(rem, digits);
            n /= 10;
        }
        if (sum == temp)
            Console.WriteLine("{0} is an Armstrong number.",
temp);
        else
            Console.WriteLine("{0} is not an Armstrong number.",
temp);
    }

    static int countDigits(int x)
    {
        int count = 0;
        while (x > 0)
        {
            count++;
            x /= 10;
        }
        return count;
    }
}
}

```

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The window has a black background with white text. The text inside the window reads: 'Enter a number to check if it is an Armstrong number or not:', '153', '153 is an Armstrong number.', and 'Press any key to continue . . .'.

```
C:\Windows\system32\cmd.exe
Enter a number to check if it is an Armstrong number or not:
153
153 is an Armstrong number.
Press any key to continue . . .
```

Q27. Write a C program to determine whether a given number is prime or not

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

namespace Question27
{
    internal class Program
    {
        static string checkPrime(int n)
        {
            if (n == 1 || n == 0)
            {
                return($"{n} is not a prime number.");
            }

            for (int i = 2; i <= Math.Sqrt(n); i++)
            {
```

```

        if (n % i == 0)
        {
            return $"{n} is not a prime number.";
        }

        return $"{n} is a prime number.";
    }

    static void Main(string[] args)
    {
        int num;
        Console.WriteLine("GIVE A NUMBER TO CHECK");
        num=Convert.ToInt32(Console.ReadLine());
        string res= checkPrime(num);
        Console.WriteLine(res);
        Console.Read();
    }
}

```

```

C:\Windows\system32\cmd.exe
GIVE A NUMBER TO CHECK
7
7 is a prime number.
_

```

Q28. Write a C program to display Pascal's triangle 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1

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

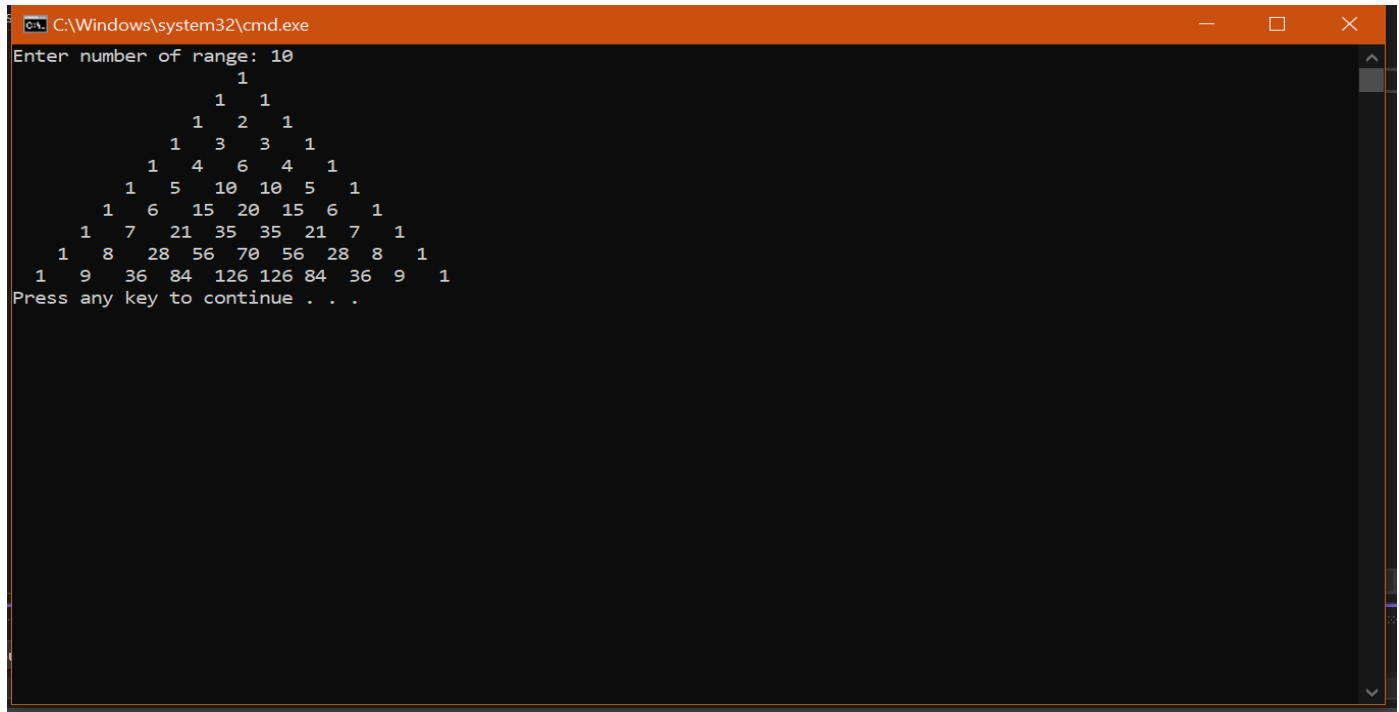
namespace Question28
{
    internal class Program
    {
        static int pascalTriangle(int i, int j)
        {
            if (j == 0 || j == i)
            {
                return 1;
            }
            else
            {
                return pascalTriangle(i - 1, j - 1) +
pascalTriangle(i - 1, j);
            }
        }
        static void Main(string[] args)
        {
            int range;
            Console.Write("Enter number of range: ");
            range = Convert.ToInt32(Console.ReadLine());

            for (int i = 0; i < range; i++)
            {
                for (int j = 0; j < range - i; j++)
                {
                    Console.Write("  ");
                }

                for (int j = 0; j <= i; j++)
                {
                    Console.Write("{0,-4}", pascalTriangle(i, j));
                }

                Console.WriteLine();
            }
        }
    }
}
```

```
}  
}
```



A screenshot of a Windows command prompt window with the title bar "C:\Windows\system32\cmd.exe". The prompt shows the output of a program that prints Pascal's Triangle for n=10. The text "Enter number of range: 10" is at the top. Below it, the triangle is displayed with 10 rows of numbers. The bottom of the window shows the prompt "Press any key to continue . . .".

```
Enter number of range: 10  
      1  
     1 1  
    1 2 1  
   1 3 3 1  
  1 4 6 4 1  
 1 5 10 10 5 1  
1 6 15 20 15 6 1  
1 7 21 35 35 21 7 1  
1 8 28 56 70 56 28 8 1  
1 9 36 84 126 126 84 36 9 1  
Press any key to continue . . .
```

Q29. Write a C program to check whether a number is a Strong Number or not

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
  
namespace Question29  
{  
    internal class Program  
    {  
        static int factorial(int x)  
        {  
            int fact=1;  
            for(int i = 1; i <= x; i++)  
            {  
                fact *= i;  
            }  
        }  
    }  
}
```

```

    }
    return fact;
}
static List<int> splitter(long num)
{
    string str = num.ToString();

    List<int> digits = new List<int>();

    for (int s = 0; s < str.Length; s++)
    {
        digits.Add(Convert.ToInt32(str[s].ToString()));
    }
    return digits;
}

static void checkStrong(int num)
{
    List<int> digits= splitter(num);
    int sum = 0;
    foreach(int s in digits)
    {
        sum += factorial(s);
    }
    if (sum == num)
    {
        Console.WriteLine($"{num} IS A STRONG
NUMBER....");
    }
    else
    {
        Console.WriteLine($"{num} IS NOT A STRONG
NUMBER....");
    }
}

static void Main(string[] args)
{
    Console.Write("GIVE NUMBER TO CHECK: ");
    int num = Convert.ToInt32(Console.ReadLine());
    checkStrong(num);
    Console.ReadKey();
}
}

```




A screenshot of a Windows Command Prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt displays two lines of text: 'GIVE NUMBER TO CHECK: 145' and '145 IS A STRONG NUMBER....'. The window has a standard Windows interface with minimize, maximize, and close buttons in the top right corner.

30. Write a program in C to get the largest element of an array using the function.

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

namespace Question30
{
    internal class Program
    {
        static int maxEle(int[] arr)
        {
            int max = arr[0];
            foreach(int ele in arr)
            {
                if (ele > max)
                {
                    max = ele;
                }
            }
            return max;
        }
    }
}
```

```

    }

    static void Main(string[] args)
    {
        int[] arr = new int[10];
        Console.WriteLine("GIVE TEN NUMBERS ONE BY ONE: ");
        for (int i = 0; i < arr.Length; i++)
        {
            arr[i] = Convert.ToInt32(Console.ReadLine());
        }
        Console.WriteLine($"THE LARGEST ELEMENT IS:
{maxEle(arr)}");
        Console.Read();
    }
}

```



```

C:\Windows\system32\cmd.exe
GIVE TEN NUMBERS ONE BY ONE:
12
32
34
46
58
70
82
94
106
118
THE BIGGEST ELEMENT IS: 118

```