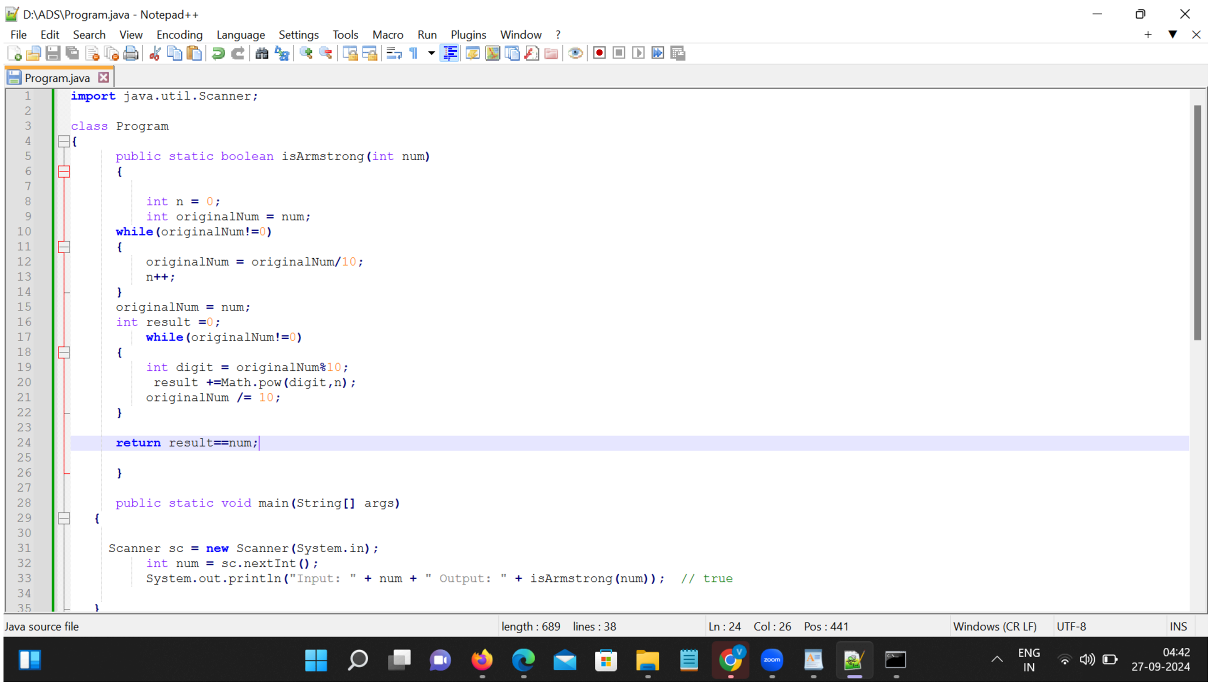
1. Armstrong Number
Problem: Write a Java program to check if a given number is an Armstrong number.

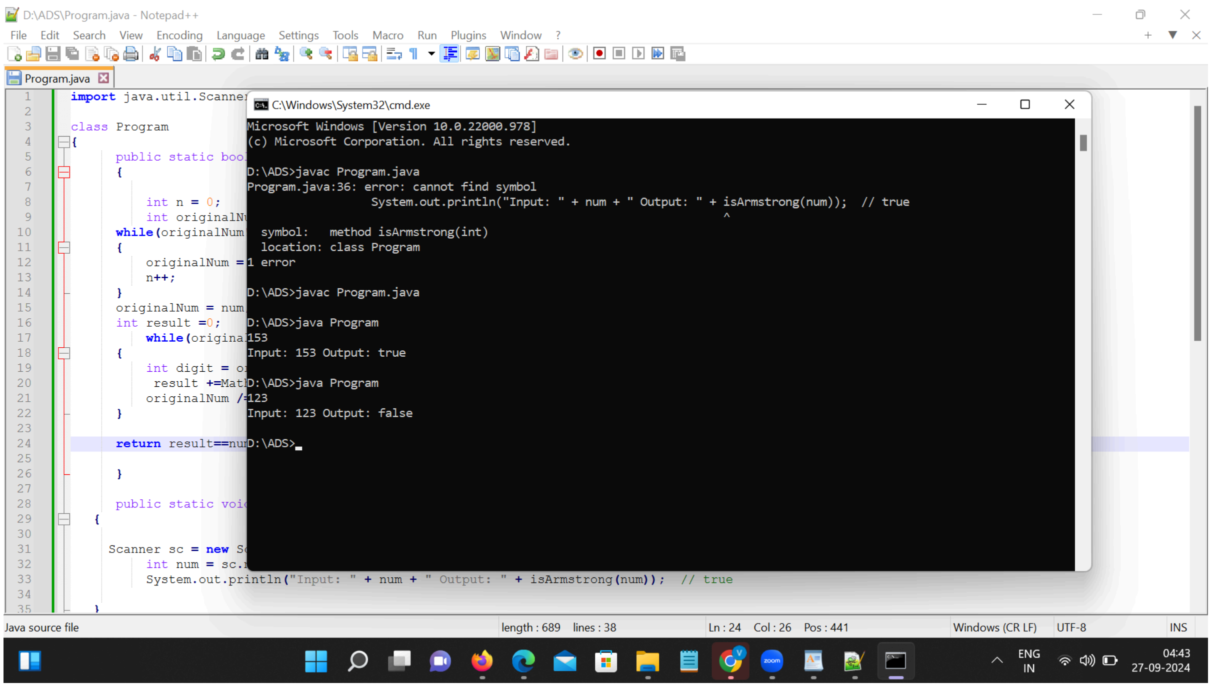
Test Cases:

Input: 153
Output: true
Input: 123
Output: false


D:\ADS\Program.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java

```java
import java.util.Scanner;

class Program
{
      public static boolean isArmstrong(int num)
      {

          int n = 0;
          int originalNum = num;
      while(originalNum!=0)
      {
          originalNum = originalNum/10;
          n++;
      }
      originalNum = num;
      int result =0;
          while(originalNum!=0)
      {
          int digit = originalNum%10;
           result +=Math.pow(digit,n);
          originalNum /= 10;
      }

      return result==num;

      }

      public static void main(String[] args)
   {

     Scanner sc = new Scanner(System.in);
          int num = sc.nextInt();
          System.out.println("Input: " + num + " Output: " + isArmstrong(num));  // true

   }
```

Java source file
length : 689 lines : 38
Ln : 24 Col : 26 Pos : 441
Windows (CR LF) UTF-8 INS
04:42
27-09-2024

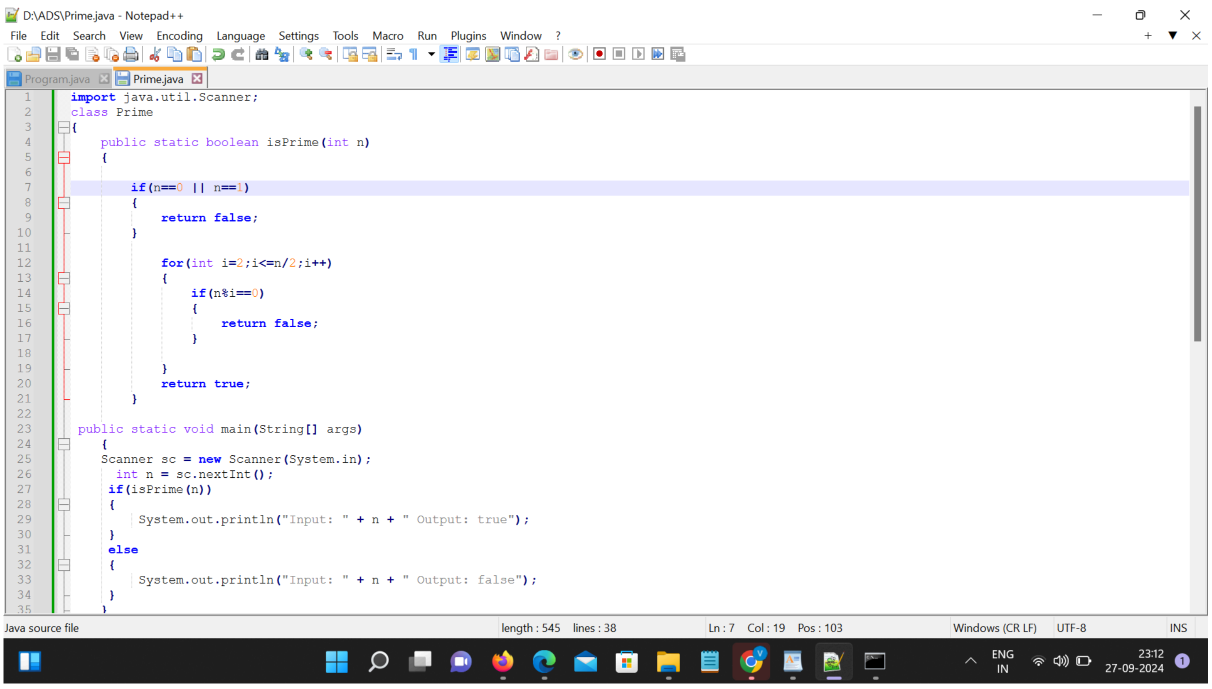

D:\ADS\Program.java - Notepad++
C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

D:\ADS>javac Program.java
Program.java:36: error: cannot find symbol
                  System.out.println("Input: " + num + " Output: " + isArmstrong(num));  // true
                                                                      ^
  symbol:   method isArmstrong(int)
  location: class Program
1 error

D:\ADS>javac Program.java

D:\ADS>java Program
153
Input: 153 Output: true

D:\ADS>java Program
123
Input: 123 Output: false

D:\ADS>_
```

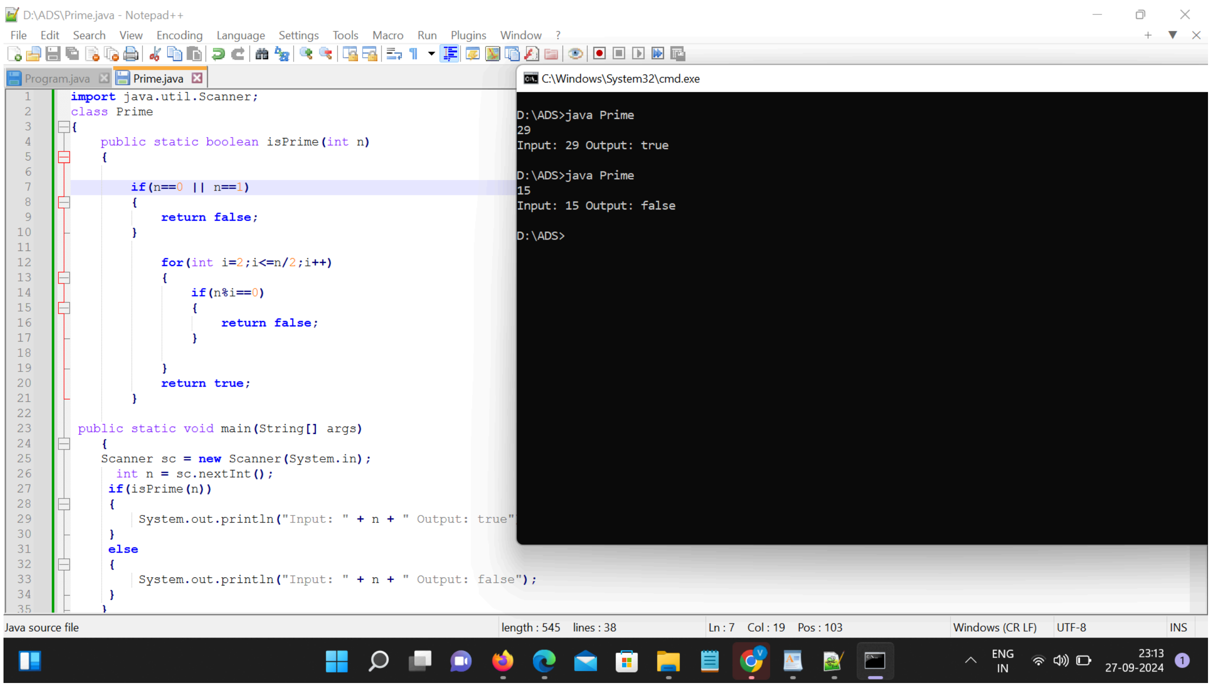
2. Prime Number
Problem: Write a Java program to check if a given number is prime.

Test Cases:

Input: 29
Output: true
Input: 15
Output: false


D:\ADS\Prime.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
Java source file
length : 545 lines : 38
Ln : 7 Col : 19 Pos : 103
Windows (CR LF) UTF-8 INS
ENG IN
23:12 27-09-2024

```java
import java.util.Scanner;
class Prime
{
    public static boolean isPrime(int n)
    {

        if(n==0 || n==1)
        {
            return false;
        }

            for(int i=2;i<=n/2;i++)
            {
                if(n%i==0)
                {
                    return false;
                }

            }
            return true;
        }

   public static void main(String[] args)
      {
      Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
       if(isPrime(n))
       {
           System.out.println("Input: " + n + " Output: true");
       }
       else
       {
           System.out.println("Input: " + n + " Output: false");
       }
      }
```



```
D:\ADS>java Prime
29
Input: 29 Output: true

D:\ADS>java Prime
15
Input: 15 Output: false

D:\ADS>
```

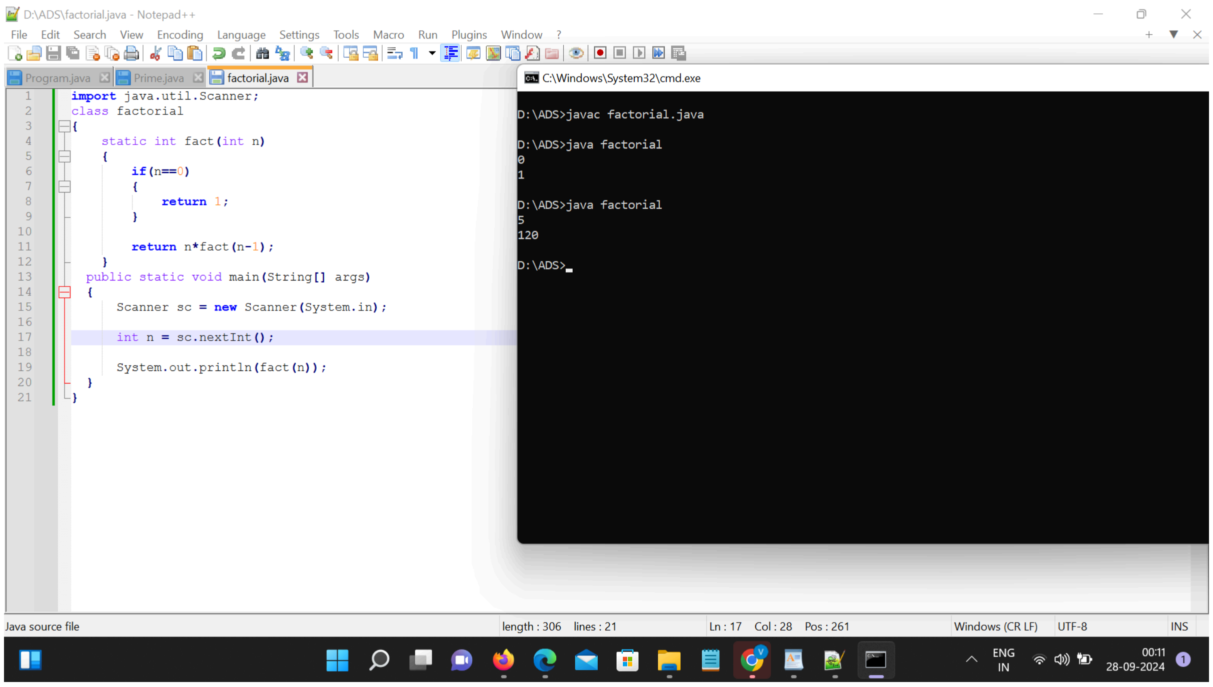
to check whether a number n is divisible by any factor, we only need to check up to n/2, because any factor greater than n/2 would pair with a factor smaller than n/2.

Time Complexity: **O(n)**
Space Complexity: **O(1)**

Factorial
Problem: Write a Java program to compute the factorial of a given number.

Test Cases:

Input: 5
Output: 120
Input: 0
Output: 1


D:\ADS\factorial.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
factorial.java
C:\Windows\System32\cmd.exe
Java source file
length : 306 lines : 21
Ln : 17 Col : 28 Pos : 261
Windows (CR LF) UTF-8 INS
ENG IN
00:11
28-09-2024

```java
import java.util.Scanner;
class factorial
{
    static int fact(int n)
    {
        if(n==0)
        {
            return 1;
        }

        return n*fact(n-1);
    }
  public static void main(String[] args)
  {
      Scanner sc = new Scanner(System.in);

      int n = sc.nextInt();

      System.out.println(fact(n));
  }
}
```

```
D:\ADS>javac factorial.java

D:\ADS>java factorial
0
1

D:\ADS>java factorial
5
120

D:\ADS>_
```

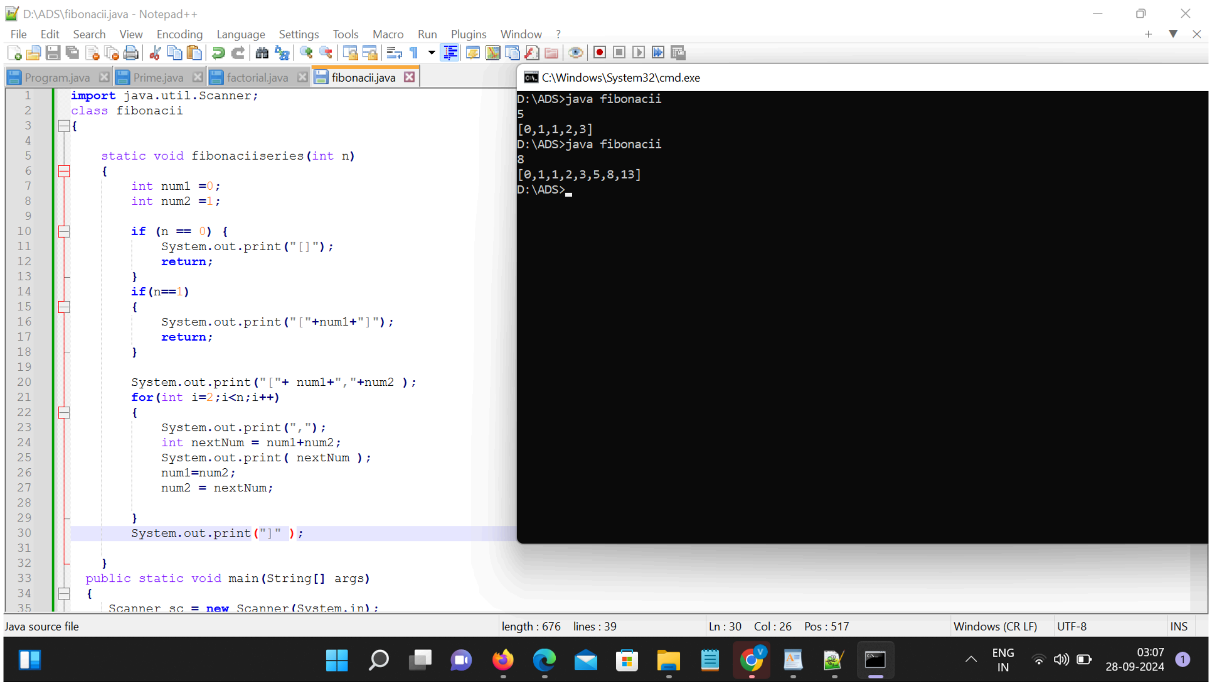
Time complexity - O(n) - The recursion proceeds until n reaches 0, meaning that the total number of recursive calls is proportional to n.

Space complexity - O(n) -Each recursive call adds a new frame to the call stack, so the space complexity is **O(n)**.

Fibonacci Series
Problem: Write a Java program to print the first n numbers in the Fibonacci series.

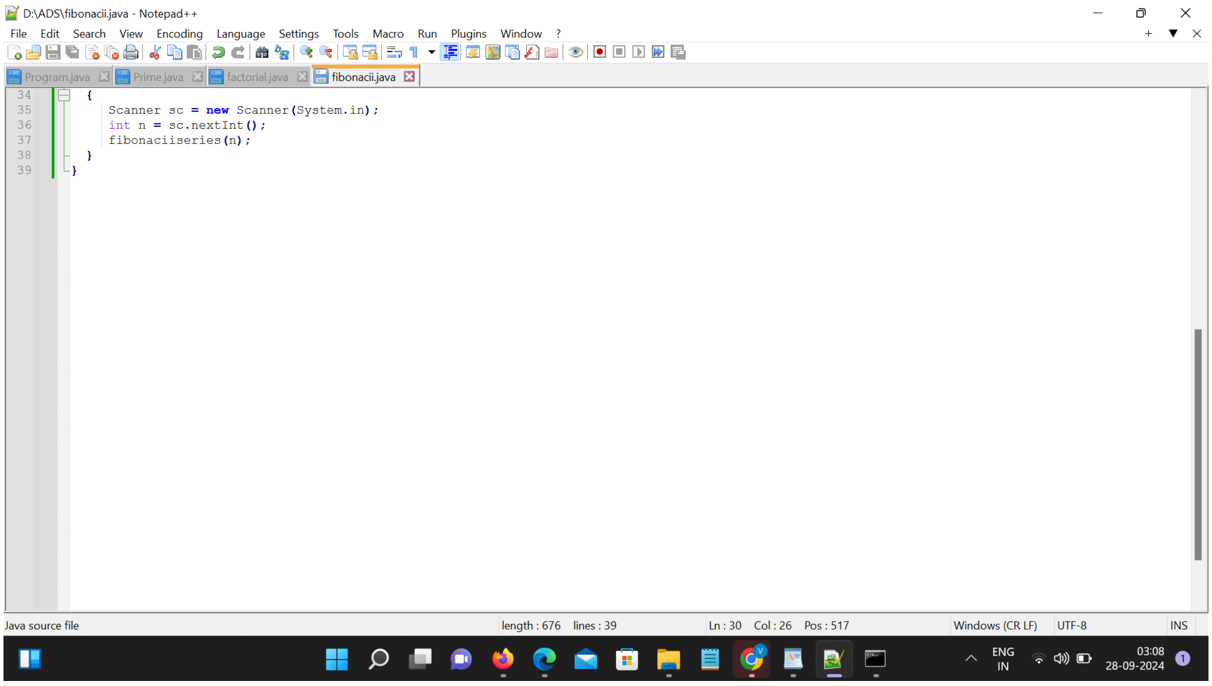
Test Cases:

Input: n = 5
Output: [0, 1, 1, 2, 3]
Input: n = 8
Output: [0, 1, 1, 2, 3, 5, 8, 13]


D:\ADS\fibonacii.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
factorial.java
fibonacii.java
C:\Windows\System32\cmd.exe
Java source file
length : 676 lines : 39
Ln : 30 Col : 26 Pos : 517
Windows (CR LF) UTF-8 INS
ENG IN
03:07 28-09-2024

```java
import java.util.Scanner;
class fibonacii
{

    static void fibonaciiseries(int n)
    {
        int num1 =0;
        int num2 =1;

        if (n == 0) {
            System.out.print("[]");
            return;
        }
        if(n==1)
        {
            System.out.print("["+num1+"]");
            return;
        }

        System.out.print("["+ num1+","+num2 );
        for(int i=2;i<n;i++)
        {
            System.out.print(",");
            int nextNum = num1+num2;
            System.out.print( nextNum );
            num1=num2;
            num2 = nextNum;

        }
        System.out.print("]" );

    }
  public static void main(String[] args)
  {
    Scanner sc = new Scanner(System.in);
```

```
D:\ADS>java fibonacii
5
[0,1,1,2,3]
D:\ADS>java fibonacii
8
[0,1,1,2,3,5,8,13]
D:\ADS>_
```


D:\ADS\fibonacii.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
factorial.java
fibonacii.java
Java source file
length : 676 lines : 39
Ln : 30 Col : 26 Pos : 517
Windows (CR LF) UTF-8 INS
ENG IN
03:08 28-09-2024

```java
  {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    fibonaciiseries(n);
  }
}
```

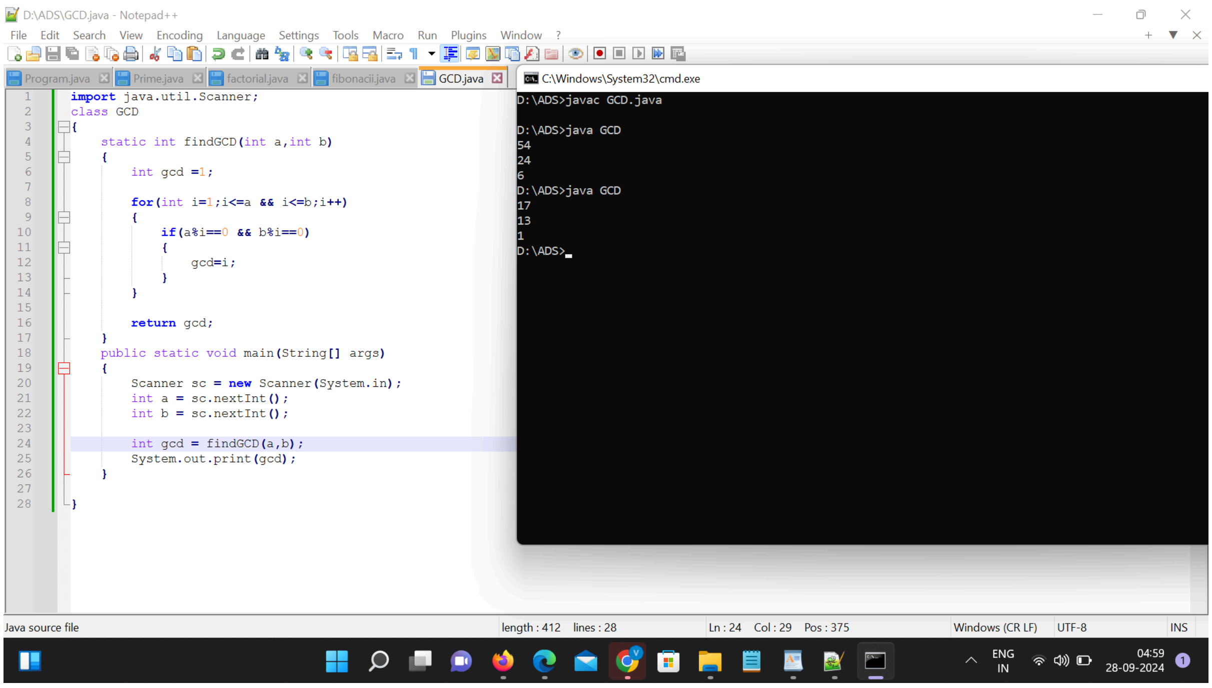
Space complexity = O(1);
Time complexity = O(n);

5. Find GCD
Problem: Write a Java program to find the Greatest Common Divisor (GCD) of two numbers.

Test Cases:

Input: a = 54, b = 24
Output: 6
Input: a = 17, b = 13

Output: 1


D:\ADS\GCD.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java Prime.java factorial.java fibonacii.java GCD.java
C:\Windows\System32\cmd.exe
Java source file length : 412 lines : 28 Ln : 24 Col : 29 Pos : 375 Windows (CR LF) UTF-8 INS
ENG IN 04:59 28-09-2024

```java
import java.util.Scanner;
class GCD
{
    static int findGCD(int a,int b)
    {
        int gcd =1;

        for(int i=1;i<=a && i<=b;i++)
        {
            if(a%i==0 && b%i==0)
            {
                gcd=i;
            }
        }

        return gcd;
    }
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        int b = sc.nextInt();

        int gcd = findGCD(a,b);
        System.out.print(gcd);
    }

}
```

```
D:\ADS>javac GCD.java

D:\ADS>java GCD
54
24
6
D:\ADS>java GCD
17
13
1
D:\ADS>_
```

Time complexity = O(1)
Space complexity = O(1)

. Find Square Root
Problem: Write a Java program to find the square root of a given number (using integer approximation).

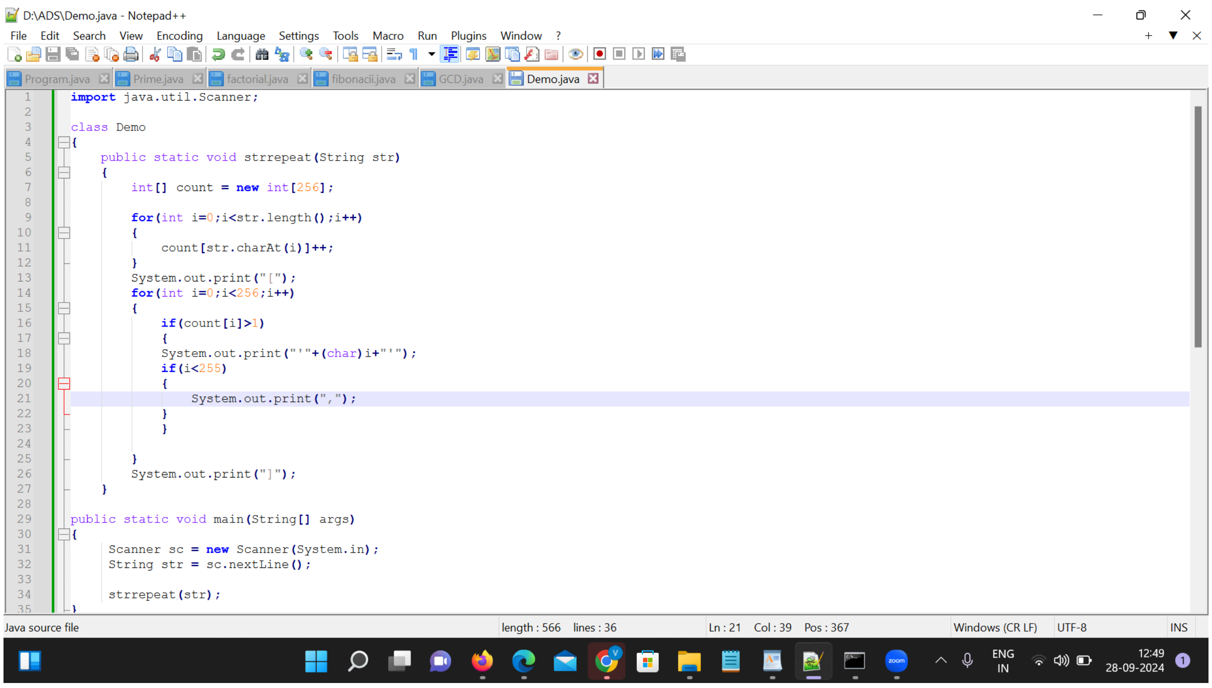
Test Cases:

Input: x = 16
Output: 4
Input: x = 27
Output: 5

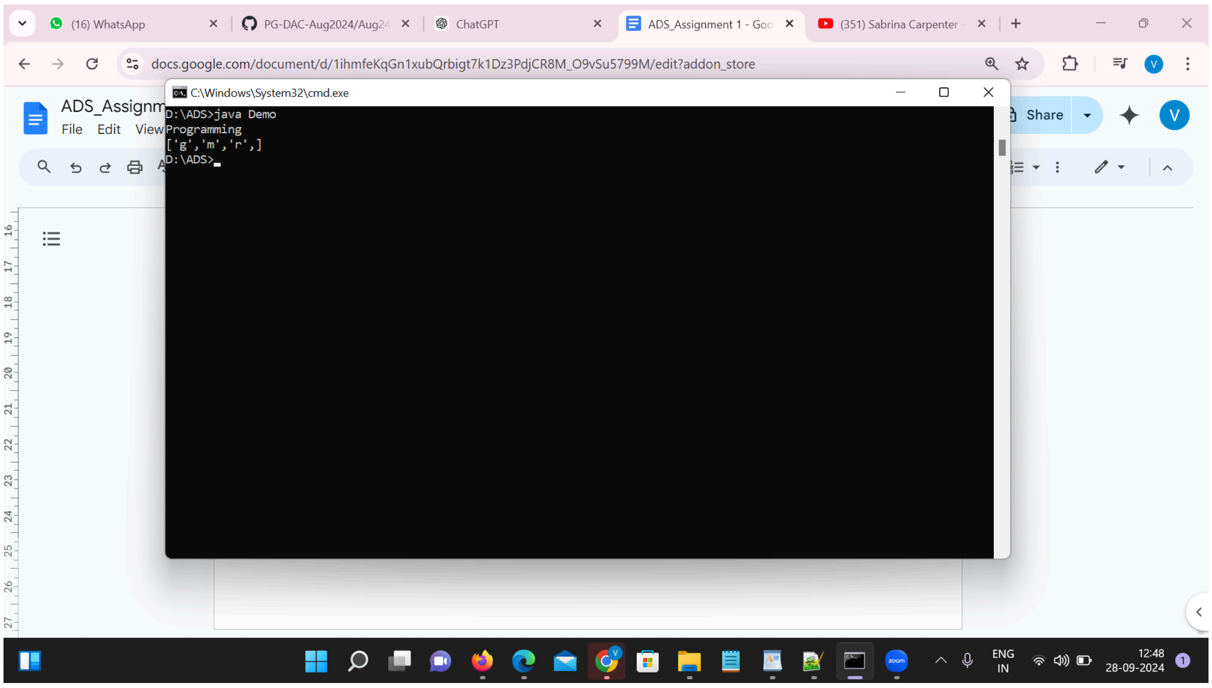
7. Find Repeated Characters in a String
Problem: Write a Java program to find all repeated characters in a string.

Test Cases:

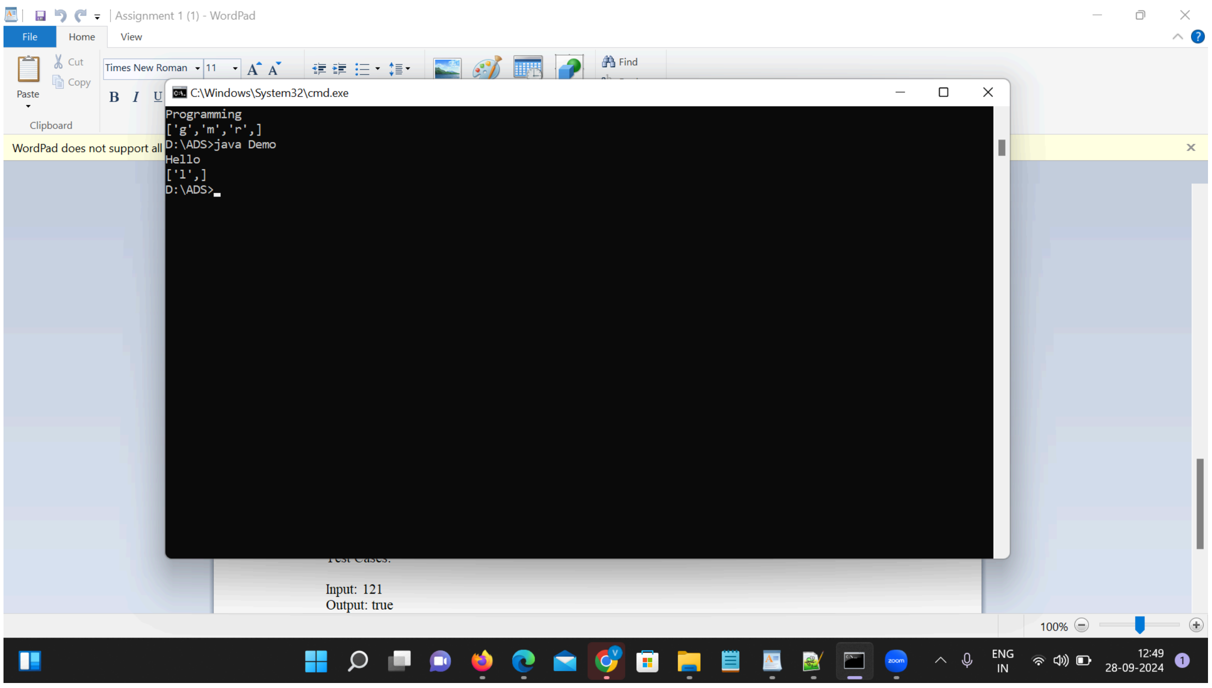
Input: "programming"
Output: ['r', 'g', 'm']
Input: "hello"
Output: ['l']


D:\ADS\Demo.java - Notepad++

```java
import java.util.Scanner;

class Demo
{
    public static void strrepeat(String str)
    {
        int[] count = new int[256];

        for(int i=0;i<str.length();i++)
        {
            count[str.charAt(i)]++;
        }
        System.out.print("[");
        for(int i=0;i<256;i++)
        {
            if(count[i]>1)
            {
            System.out.print("'"+(char)i+"'");
            if(i<255)
            {
                System.out.print(",");
            }
            }

        }
        System.out.print("]");
    }

public static void main(String[] args)
{
    Scanner sc = new Scanner(System.in);
    String str = sc.nextLine();

    strrepeat(str);
}
```


C:\Windows\System32\cmd.exe

```
D:\ADS>java Demo
Programming
['g','m','r',]
D:\ADS>
```



```
Programming
['g','m','r',]
D:\ADS>java Demo
Hello
['l',]
D:\ADS>_
```

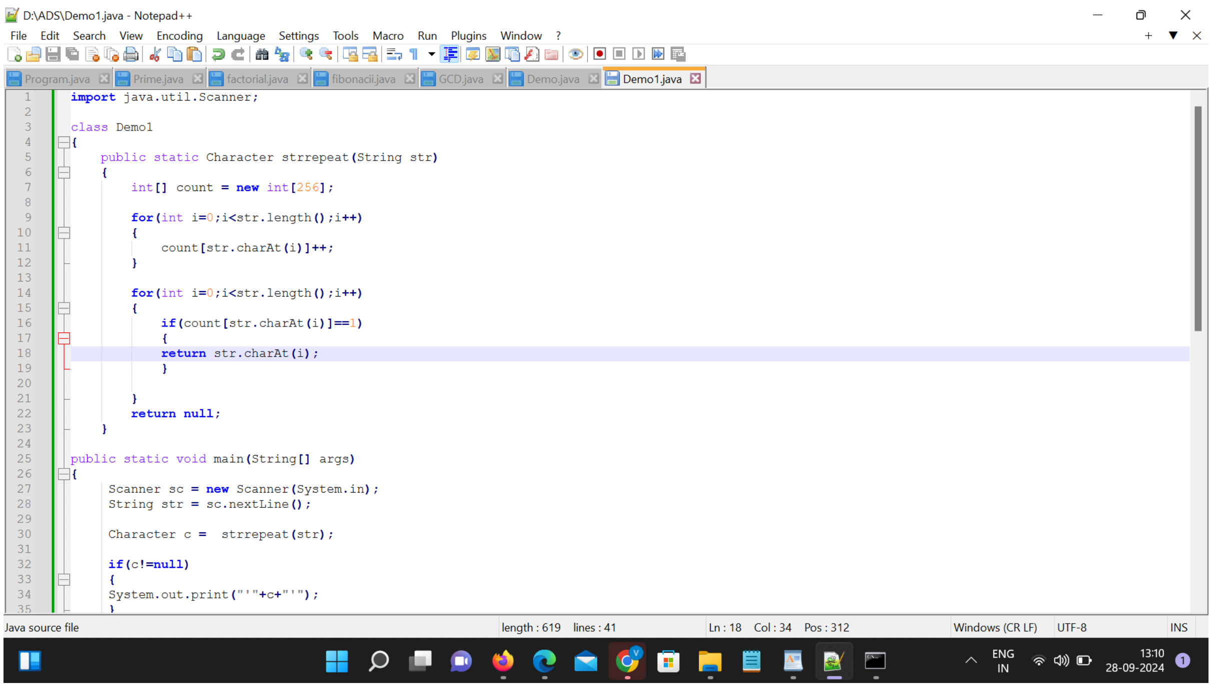
Input: 121
Output: true

Space complexity = O(n)
Time complexity = O(n)

8. First Non-Repeated Character
Problem: Write a Java program to find the first non-repeated character in a string.

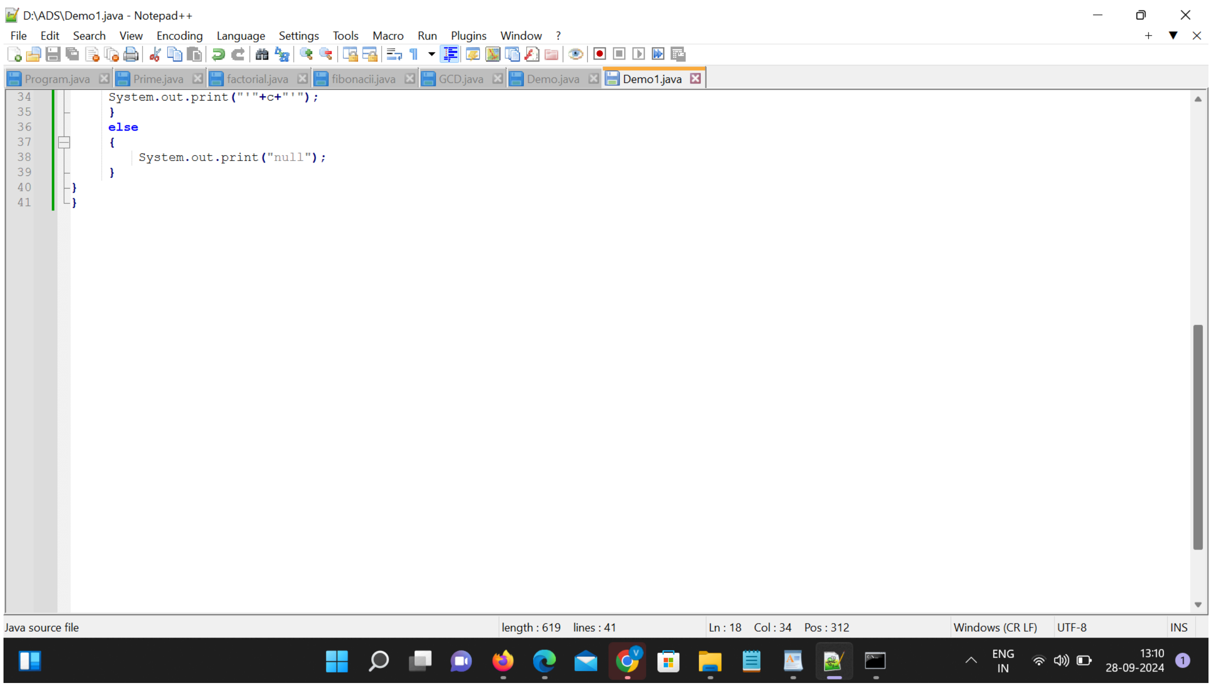
Test Cases:

Input: "stress"
Output: 't'
Input: "aabbcc"
Output: null


D:\ADS\Demo1.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
factorial.java
fibonacii.java
GCD.java
Demo.java
Demo1.java

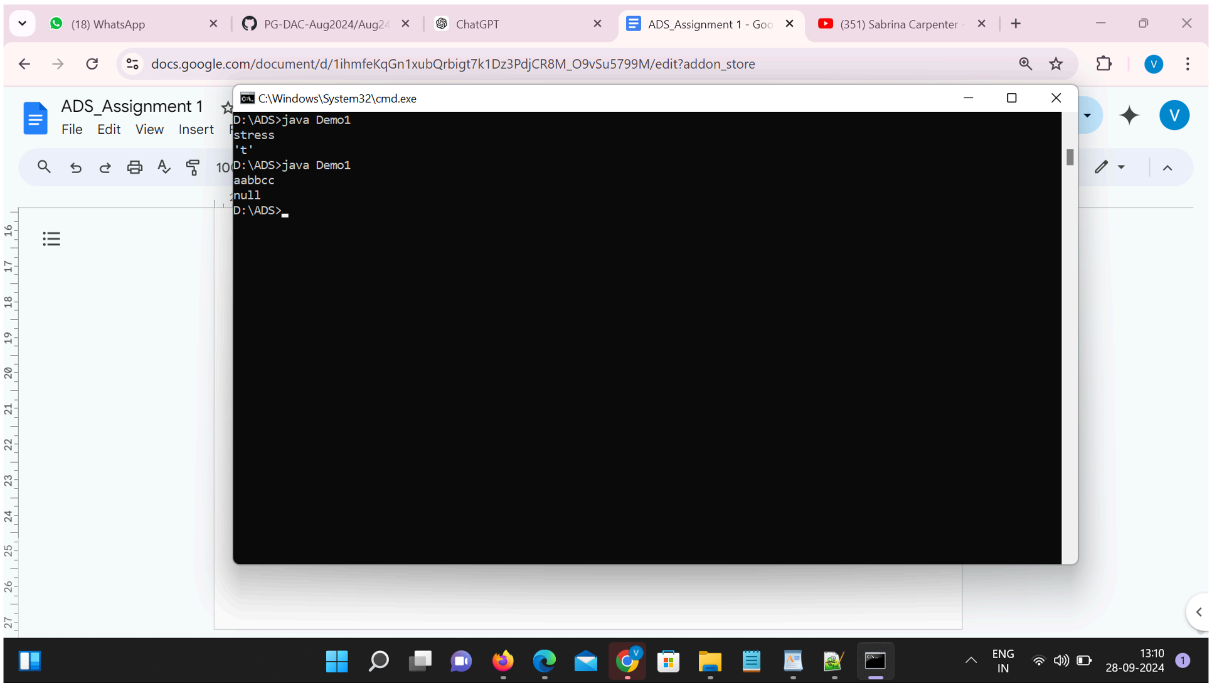
```java
import java.util.Scanner;

class Demo1
{
    public static Character strrepeat(String str)
    {
        int[] count = new int[256];

        for(int i=0;i<str.length();i++)
        {
            count[str.charAt(i)]++;
        }

        for(int i=0;i<str.length();i++)
        {
            if(count[str.charAt(i)]==1)
            {
            return str.charAt(i);
            }

        }
        return null;
    }

public static void main(String[] args)
{
     Scanner sc = new Scanner(System.in);
     String str = sc.nextLine();

     Character c =  strrepeat(str);

     if(c!=null)
     {
     System.out.print("'"+c+"'");
     }
```

Java source file
length : 619   lines : 41
Ln : 18   Col : 34   Pos : 312
Windows (CR LF)   UTF-8   INS
ENG IN
13:10
28-09-2024


D:\ADS\Demo1.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Program.java
Prime.java
factorial.java
fibonacii.java
GCD.java
Demo.java
Demo1.java

```java
     System.out.print("'"+c+"'");
     }
     else
     {
         System.out.print("null");
     }
}
}
```

Java source file
length : 619   lines : 41
Ln : 18   Col : 34   Pos : 312
Windows (CR LF)   UTF-8   INS
ENG IN
13:10
28-09-2024



```
D:\ADS>java Demo1
stress
't'
D:\ADS>java Demo1
aabbcc
null
D:\ADS>
```