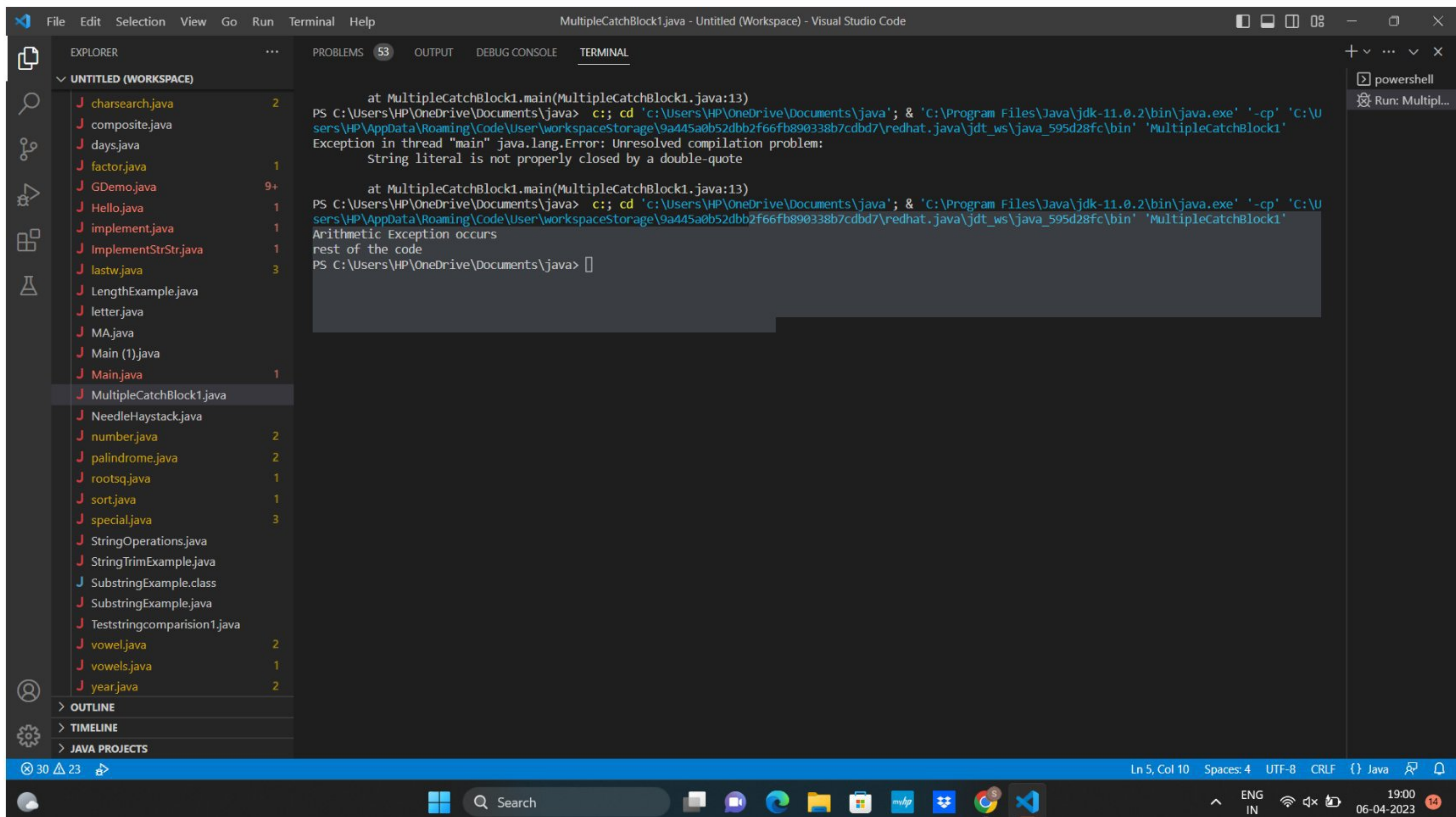


PROGRAM:1

```
public class MultipleCatchBlock1 {  
    public static void main(String[] args) {  
        try{  
            int a[]=new int[5];  
            a[5]=30/0;  
        }  
        catch(ArithmeticException e)  
        {  
            System.out.println("Arithmetic Exception occurs");  
        }  
        catch(ArrayIndexOutOfBoundsException e)  
        {  
            System.out.println("ArrayIndexOut Of Bounds Exceptionoccurs");  
        }  
        catch(Exception e)  
        {  
            System.out.println("Parent Exception occurs");  
        }  
        System.out.println("rest of the code");  
    }  
}
```

OUTPUT:



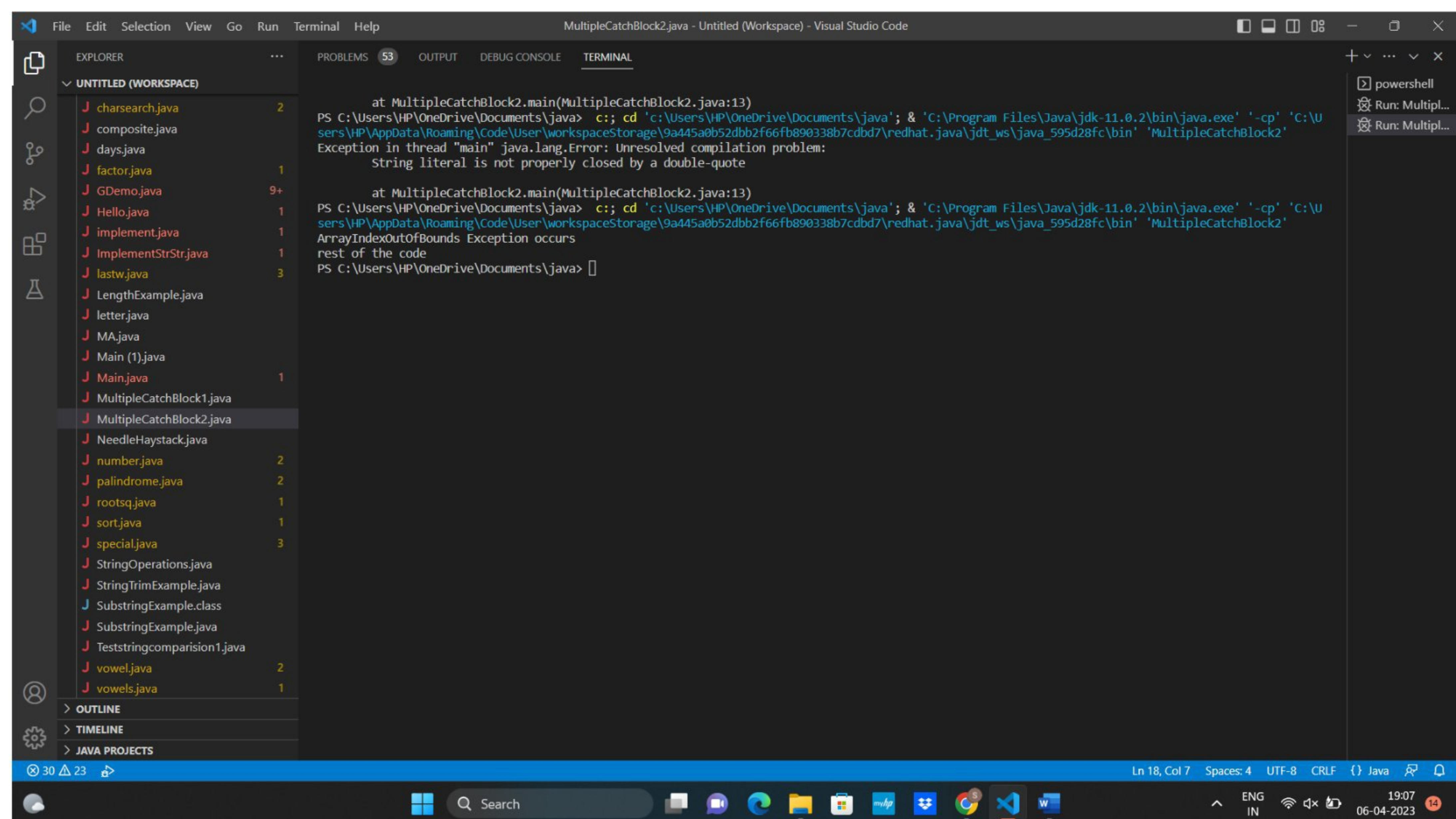
The screenshot shows the Visual Studio Code interface with the 'MultipleCatchBlock1.java' file open. The 'TERMINAL' panel is active, displaying the output of the program. The output shows the execution of the program, which prints 'Arithmetic Exception occurs' and 'rest of the code'.

```
at MultipleCatchBlock1.main(MultipleCatchBlock1.java:13)  
PS C:\Users\HP\OneDrive\Documents\java> c:: cd 'c:\Users\HP\OneDrive\Documents\java'; & 'C:\Program Files\Java\jdk-11.0.2\bin\java.exe' '-cp' 'C:\U  
sers\HP\AppData\Roaming\Code\User\workspaceStorage\9a445a0b52dbb2f66fb890338b7cbbd7\redhat.java\jdt_ws\java_595d28fc\bin' 'MultipleCatchBlock1'  
Exception in thread "main" java.lang.Error: Unresolved compilation problem:  
String literal is not properly closed by a double-quote  
at MultipleCatchBlock1.main(MultipleCatchBlock1.java:13)  
PS C:\Users\HP\OneDrive\Documents\java> c:: cd 'c:\Users\HP\OneDrive\Documents\java'; & 'C:\Program Files\Java\jdk-11.0.2\bin\java.exe' '-cp' 'C:\U  
sers\HP\AppData\Roaming\Code\User\workspaceStorage\9a445a0b52dbb2f66fb890338b7cbbd7\redhat.java\jdt_ws\java_595d28fc\bin' 'MultipleCatchBlock1'  
Arithmetic Exception occurs  
rest of the code  
PS C:\Users\HP\OneDrive\Documents\java> 
```


ARRAY INDEX OUT OF BOUNDS:

```
public class MultipleCatchBlock2 {
    public static void main(String[] args)
    {
        try
        {
            int a[]=new int[5];
            System.out.println(a[10]);
        }
        catch(ArithmeticException e)
        {
            System.out.println("Arithmetic Exception occurs");
        }
        catch(ArrayIndexOutOfBoundsException e)
        {
            System.out.println("ArrayIndexOutOfBounds Exception occurs");
        }
        catch(Exception e)
        {
            System.out.println("Parent Exception occurs");
        }
        System.out.println("rest of the code");
    }
}
```

OUTPUT:



NULL POINTER EXCEPTIPON:


```

import java.io.*;
class ne
{
public static void main (String[] args)
{
// Initializing String variable with null value
String ptr = null;
// Checking if ptr.equals null or works fine.
try
{
if (ptr.equals("gfg"))
System.out.print("Same");
else
System.out.print("Not Same");
}
catch (NullPointerException e)
{
System.out.print("NullPointerException Caught");
}
}
}
}

```

OUTPUT:

PROGRAM:2

```

class Table
{

```



```

void printTable(int n)
{
    synchronized(this)
    {
        for(int i=1;i<=5;i++)
        {
            System.out.println(+n+"*"+i+"="+n*i);
            try
            {
                Thread.sleep(400);
            }
            catch(Exception e)
            {
                System.out.println(e);
            }
        }
    }
}

class Mythread1 extends Thread
{
    Table t;
    Mythread1(Table t)
    {
        this.t=t;
    }
    public void run()
    {
        t.printTable(5);
    }
}

class Mythread2 extends Thread
{
    Table t;
    Mythread2(Table t)
    {
        this.t=t;
    }
    public void run()
    {
        t.printTable(100);
    }
}

class Use
{
    public static void main(String args[])
    {
        Table obj = new Table();
    }
}

```

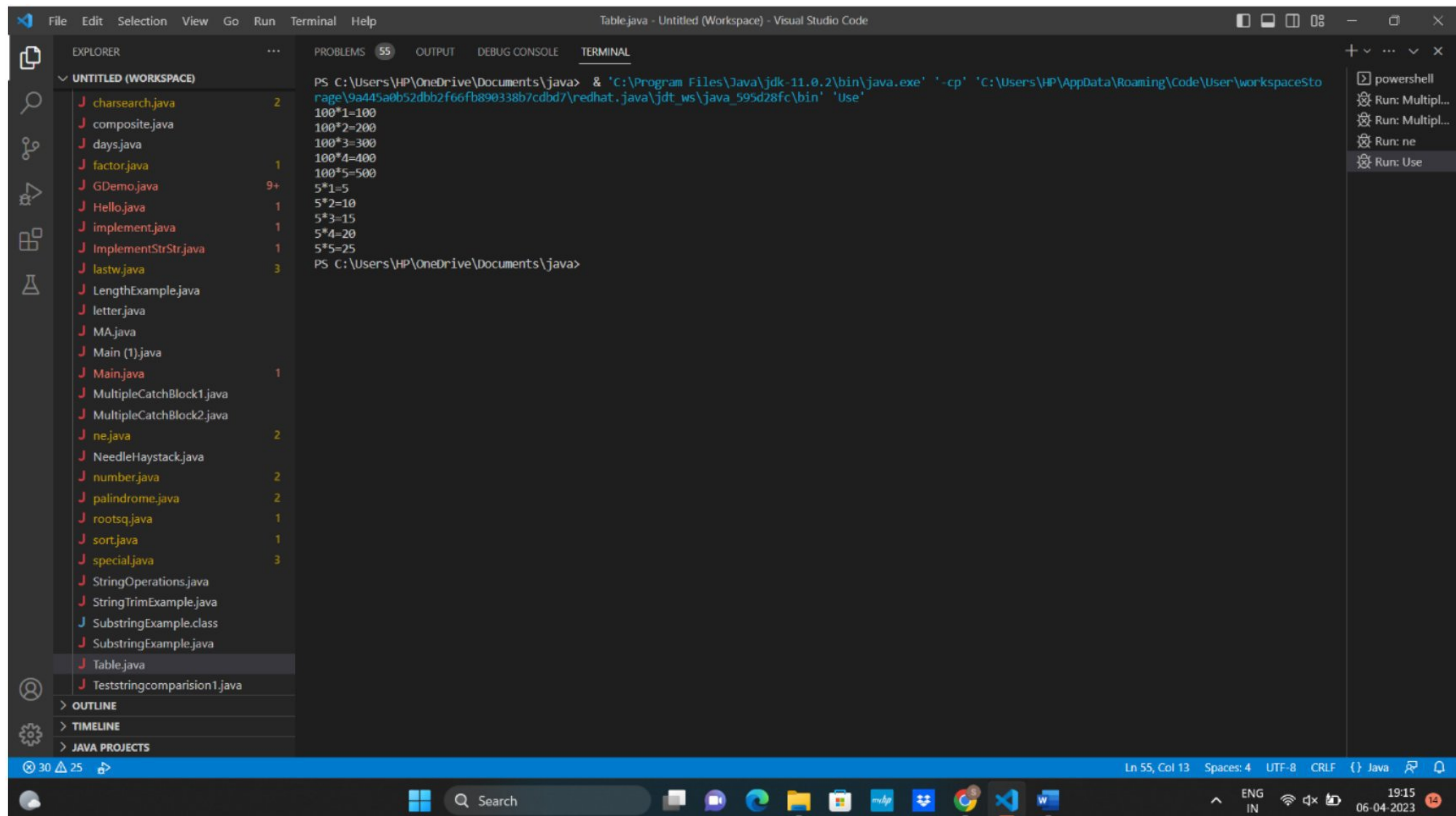


```

Mythread1 th1 = new Mythread1(obj);
Mythread2 th2 = new Mythread2(obj);
th1.start();
th2.start();
}
}

```

OUTPUT:



PROGRAM:3

```

import java.util.*;
import java.io.*;
public class ugly {
    public static void main(String args[]) {
        int inputNumber;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the number :");
        inputNumber=sc.nextInt();
        boolean check = true;
        for(int i = 2; i<=inputNumber; i++) {
            if(i!=2&&i!=3&&i!=5) {
                if(inputNumber%i==0&&checkPrime(i)) {
                    check = false;
                    break;
                }
            }
        }
        if(check) {

```



```

System.out.println(inputNumber+" is an ugly number");
} else {
System.out.println(inputNumber+" is Not an ugly number");
}
}
static boolean checkPrime(int number)
{
    boolean flag = true;
    for(int i = 2; i<=number/2; i++) {
    if(number%i==0) {
    flag = false;
    break;
    }
    }
    return flag;
}
}

```

OUTPUT:

The screenshot shows the Visual Studio Code interface. The Explorer pane on the left lists various Java files, with 'ugly.java' selected. The Terminal pane on the right shows the output of running the program in a PowerShell window. The output indicates that the program is running successfully and has identified the input number 15 as an ugly number.

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\HP\OneDrive\Documents\java> & "C:\Program Files\Java\jdk-11.0.2\bin\java.exe" "-cp" "C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\9a445a0b524bb2f66fb890338b7c0bd7\redhat.java\jdk_ws\java_595d28fc\bin" "ugly"
Enter the number :
15
15 is an ugly number
PS C:\Users\HP\OneDrive\Documents\java>

```

PROGRAM:4

```

import java.io.*;
import java.util.*;
class fibo
{
static int fib(int n)
{

```

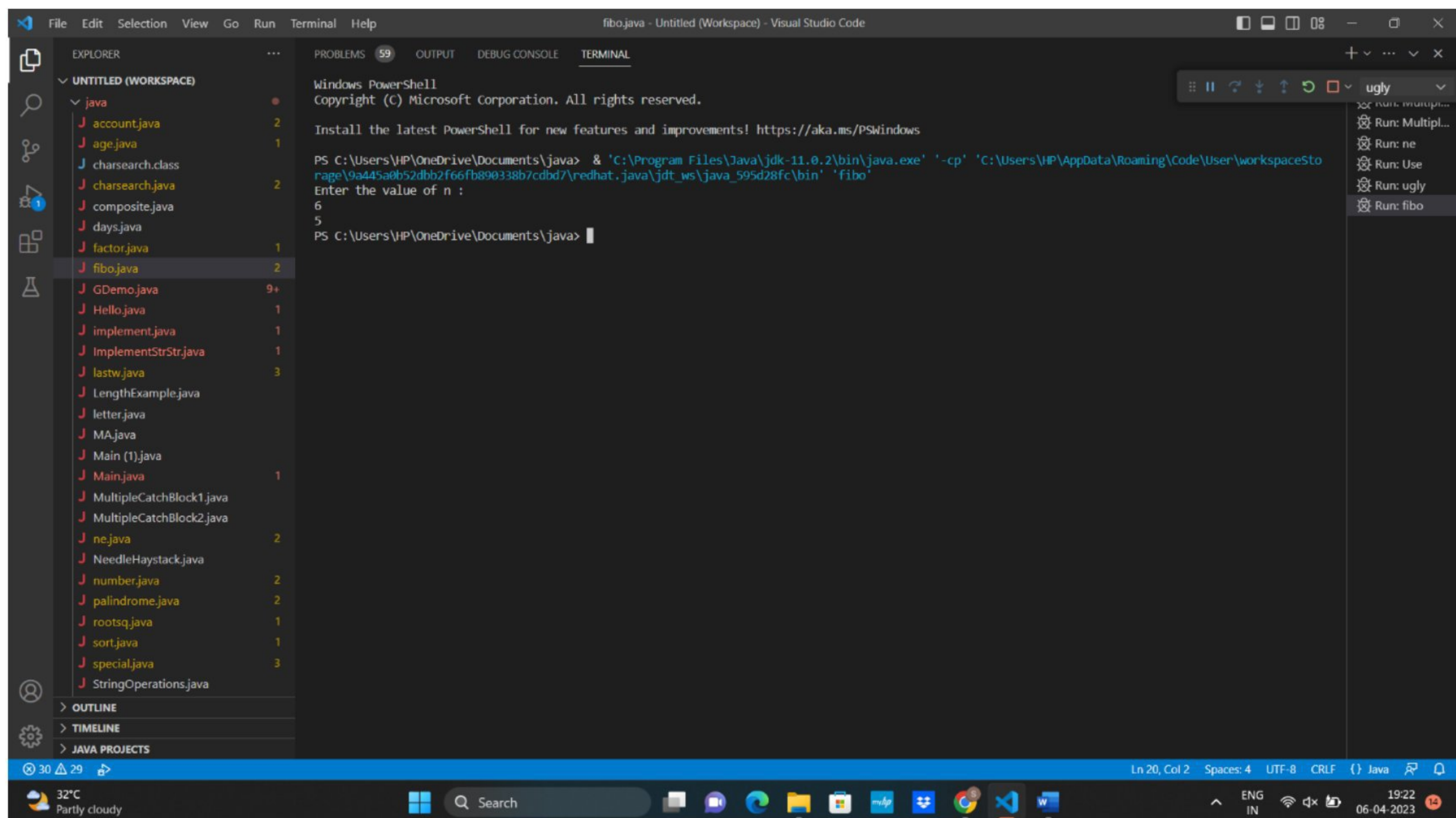


```

if (n==0||n==1)
return 0;
else if(n==2)
return 1;
return fib(n - 1) + fib(n - 2);
}
public static void main(String args[])
{
int n;
Scanner sc=new Scanner(System.in);
System.out.println("Enter the value of n : ");
n=sc.nextInt();
System.out.println(fib(n));
}
}

```

OUTPUT:



PROGRAM:5

```

import java.io.*;
import java.util.*;
class duplicate
{
static int removeDuplicates(int arr[], int n)
{
if (n == 0 || n == 1)
return n;
int[] temp = new int[n];

```



```

int j = 0;
for (int i = 0; i < n-1; i++)
{
    if (arr[i] != arr[i+1])
temp[j++] = arr[i];
}
temp[j++] = arr[n-1];
for (int i = 0; i < j; i++) {
arr[i] = temp[i];
}
return j;
}

public static void main(String[] args) {
int arr[] = {10, 20, 20, 30, 40, 40, 40, 50, 50};
int n = arr.length;
n = removeDuplicates(arr, n);
for (int i = 0; i < n; i++) {
System.out.print(arr[i]+" ");
}
}
}

```

OUTPUT:

The screenshot shows the Visual Studio Code interface. The Explorer pane on the left displays a list of Java files in a project named 'untitled (workspace)'. The file 'duplicate.java' is selected. The Terminal pane on the right shows the output of the program, which is '10 20 30 40 50'. The status bar at the bottom indicates the current file is 'duplicate.java' at line 26, column 5.