Lab Exercise 4

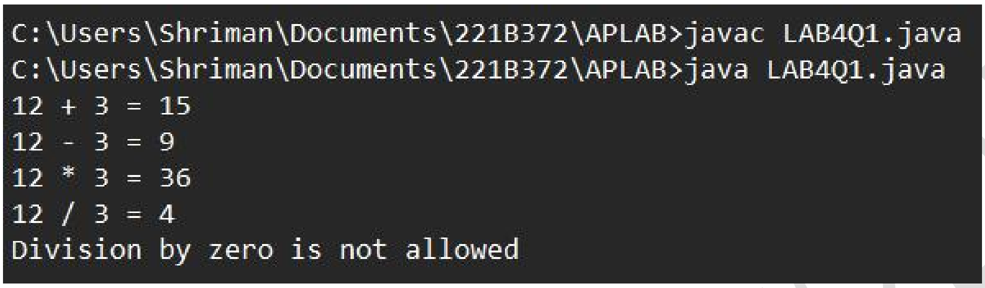
1. You are developing a simple calculator program. Implement a method calculate(int num1, int num2, String operator) that performs basic arithmetic operations (addition, subtraction, multiplication, division). Handle exceptions for invalid operators and division by zero. Use custom exceptions where appropriate.

```
public class LAB4Q1 {

  public static int calculate(int num1, int num2, String operator)

  {if (operator.equals("+")) {

     return num1 + num2;      }

 else if (operator.equals("-")) {

    return num1 - num2;

     }

else if (operator.equals("*"))

     { return num1 * num2;     }

else if (operator.equals("/"))

 {if (num2 == 0) {

        throw new ArithmeticException("Division by zero is not allowed.");

       }

       return num1 / num2;

     } else {

        throw new IllegalArgumentException("Invalid operator. Use +, -, *, or /.");      }

  }

  public static void main(String[] args)

    {try {

       System.out.println("10 + 5 = " + calculate(10, 5, "+"));

       System.out.println("10 - 5 = " + calculate(10, 5, "-"));

       System.out.println("10 * 5 = " + calculate(10, 5, "*"));

       System.out.println("10 / 5 = " + calculate(10, 5, "/"));

       System.out.println("10 / 0 = " + calculate(10, 0, "/")); // Expected to throw
ArithmeticException
```

SHRIMAN VIJAY(221B372)

```
            System.out.println("10 % 5 = " + calculate(10, 5, "%")); // Expected to throw
IllegalArgumentException

        } catch (ArithmeticException | IllegalArgumentException e) {

            System.out.println(e.getMessage());

        }

    }

}
```

OUTPUT



```
C:\Users\Shriman\Documents\221B372\APLAB>javac LAB4Q1.java
C:\Users\Shriman\Documents\221B372\APLAB>java LAB4Q1.java
12 + 3 = 15
12 - 3 = 9
12 * 3 = 36
12 / 3 = 4
Division by zero is not allowed
```

2. Design a class TemperatureConverter with methods to convert between Fahrenheit and Celsius temperatures. Implement error handling for invalid temperature values (e.g., below absolute zero). Use custom exceptions to report temperature-related errors.

```
public class LAB4Q2 {
    static class TemperatureOutOfRangeException extends Exception
   {private String message;

        public TemperatureOutOfRangeException(String message)
             {this.message = message;

        }

        @Override public
String  getMessage() {

      return message;
      }
```

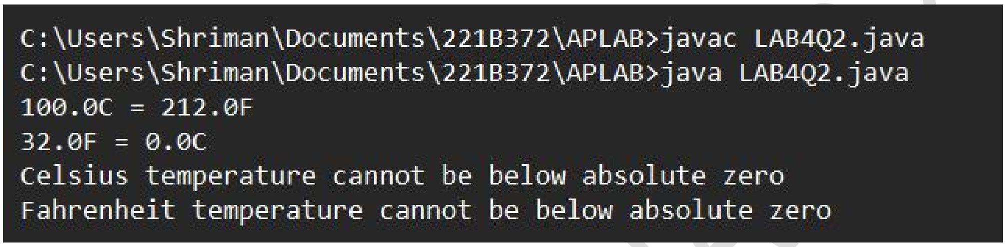
SHRIMAN VIJAY(221B372)

```
   }

   public static double celsiusToFahrenheit(double celsius) throws
TemperatureOutOfRangeException {
       if (celsius < -273.15) {

           throw new TemperatureOutOfRangeException("Celsius temperature cannot be below
absolute zero (-273.15°C).");

       }

  return (celsius * 9 / 5) +  32;

   }

   public static double fahrenheitToCelsius(double fahrenheit) throws
TemperatureOutOfRangeException {

        if (fahrenheit < -459.67) {

        throw new TemperatureOutOfRangeException("Fahrenheit temperature cannot be
below absolute zero (-459.67°F).");

       }

       return (fahrenheit - 32) * 5 / 9;

   }

   public static void main(String[] args)

       {try {

   double celsius = 100;

  double fahrenheit = celsiusToFahrenheit(celsius);

           System.out.println(celsius + "°C = " + fahrenheit + "°F");

           fahrenheit = 32;

           celsius = fahrenheitToCelsius(fahrenheit);

           System.out.println(fahrenheit + "°F = " + celsius + "°C");

           celsius = -300;

          fahrenheit = celsiusToFahrenheit(celsius);

       }

catch (TemperatureOutOfRangeException e) {
```

SHRIMAN VIJAY(221B372)

```
            System.out.println(e.getMessage());
        }
  try {
   double fahrenheit = -500;
  double celsius = fahrenheitToCelsius(fahrenheit);
        }
catch (TemperatureOutOfRangeException e)
            {System.out.println(e.getMessage());
        } } }
```

OUTPUT:



```
C:\Users\Shriman\Documents\221B372\APLAB>javac LAB4Q2.java
C:\Users\Shriman\Documents\221B372\APLAB>java LAB4Q2.java
100.0C = 212.0F
32.0F = 0.0C
Celsius temperature cannot be below absolute zero
Fahrenheit temperature cannot be below absolute zero
```

3. Create a program to manage a library catalog. Implement a Book class with attributes like title, author, and availableCopies. Develop a method to check out book, which may throw a custom exception BookUnavailableException when no copies are available. Implement proper exception handling in the library catalog system.

```
class BookUnavailableException extends Exception
    { public BookUnavailableException(String message) {
        super(message);
    }
}
class Book {
    private String title;
    private String author;
    private int availableCopies;
```

SHRIMAN VIJAY (221B372)

```
  public Book(String title, String author, int availableCopies)
    {this.title = title;
    this.author = author;
    this.availableCopies = availableCopies;
  }
public String getTitle() {
    return title;
  }
  public String getAuthor()
    {return author;
  }
  public int getAvailableCopies()
    {return availableCopies;
  }
  public void checkOut() throws BookUnavailableException
    {if (availableCopies > 0) {
      availableCopies--;
      System.out.println("Successfully checked out " + title);
    } else {
      throw new BookUnavailableException("No copies of " + title + " are available.");
    }
  }
  public void addCopy() {
    availableCopies++;
  }
```

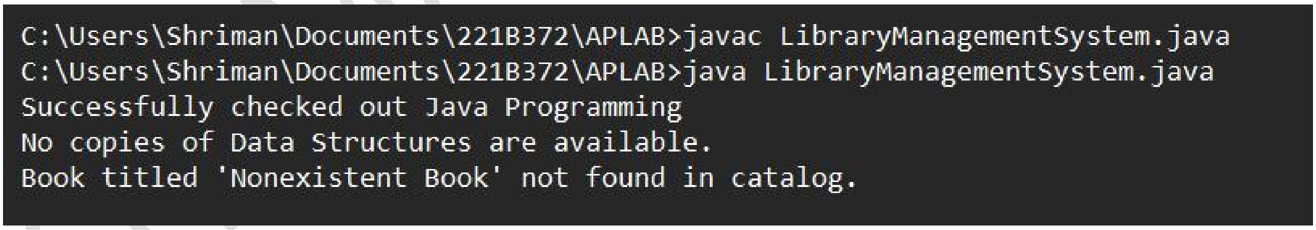
SHRIMAN VIJAY(221B372)

```
}
class LibraryCatalog {
   private List<Book> catalog;


   public LibraryCatalog()
      { catalog = new
      ArrayList<>();
   }
   public void addBook(Book book) {
      catalog.add(book);
   }
   public Book findBook(String title) {
      for (Book book : catalog) {
         if (book.getTitle().equalsIgnoreCase(title))
            {return book;
         }
      }
      return null;
   }
   public void checkOutBook(String title) {
      Book book = findBook(title);
      if (book != null) {
         try {
            book.checkOut();
         } catch (BookUnavailableException e)
            {System.out.println(e.getMessage());
```

SHRIMAN VIJAY(221B372)

```
            }
        } else {
            System.out.println("Book titled '" + title + "' not found in catalog.");
        }
    }
 }
public class LibraryManagementSystem {
    public static void main(String[] args) {
        LibraryCatalog catalog = new LibraryCatalog();


        catalog.addBook(new Book("Java Programming", "Author A", 3));
        catalog.addBook(new Book("Data Structures", "Author B", 0));


        catalog.checkOutBook("Java Programming"); // Should succeed
        catalog.checkOutBook("Data Structures");  // Should throw BookUnavailableException
        catalog.checkOutBook("Nonexistent Book"); // Should print not found message
    }
 }
```

OUTPUT :



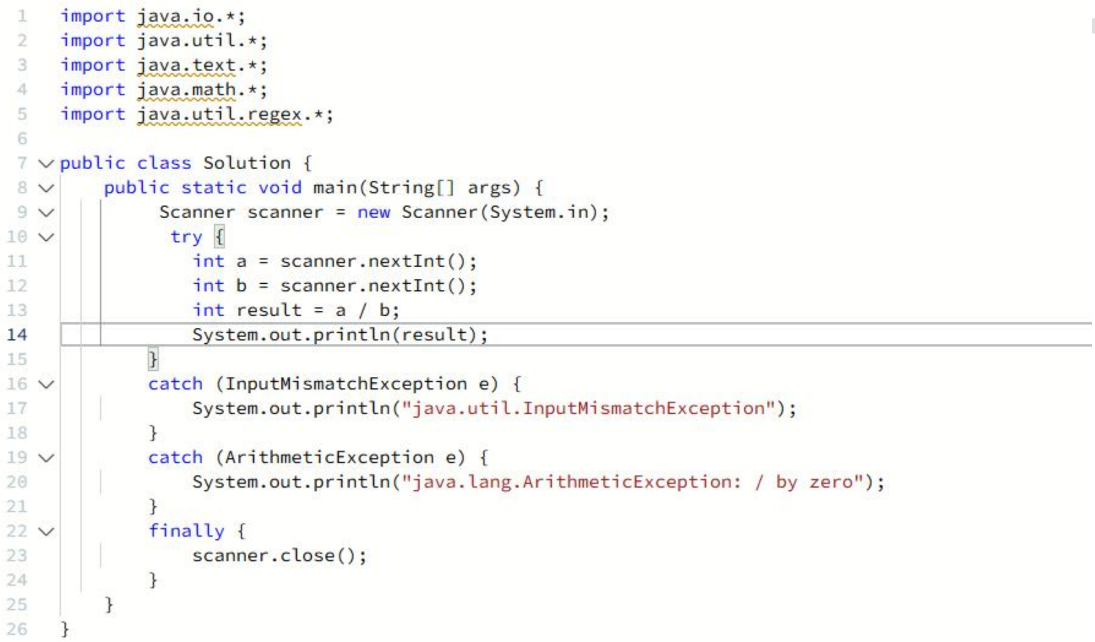
```
C:\Users\Shriman\Documents\221B372\APLAB>javac LibraryManagementSystem.java
C:\Users\Shriman\Documents\221B372\APLAB>java LibraryManagementSystem.java
Successfully checked out Java Programming
No copies of Data Structures are available.
Book titled 'Nonexistent Book' not found in catalog.
```

SHRIMAN VIJAY(221B372)

4. Solve the following questions:

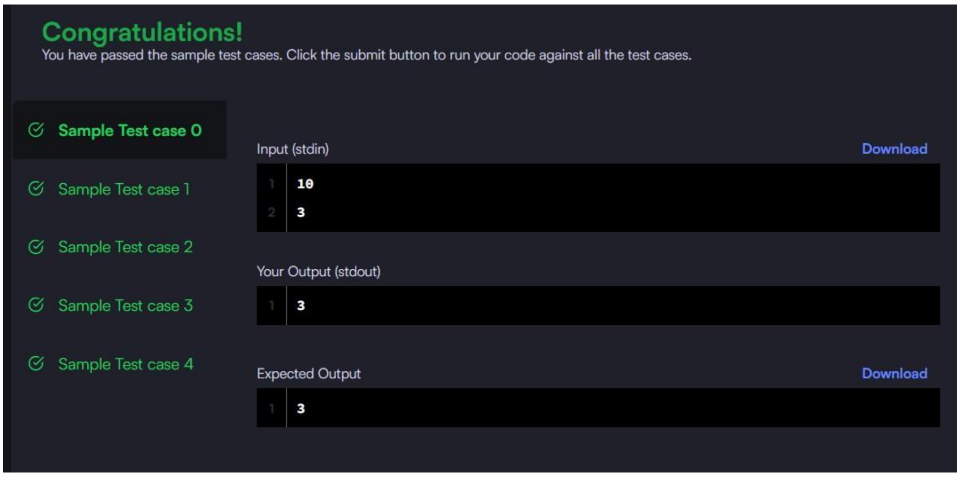
a. **https://www.hackerrank.com/challenges/java-exception-handling-try-catch/problem**
b. **https://www.hackerrank.com/challenges/java-exception-handling/problem?isFullScreen=true**
c. **https://www.hackerrank.com/challenges/simple-addition-varargs/problem?isFullScreen=true**
d. **https://www.hackerrank.com/challenges/java-1d-array-introduction/problem**
e. **https://www.hackerrank.com/challenges/java-2d-array/problem?isFullScreen=true**

a)



```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {
    public static void main(String[] args) {
         Scanner scanner = new Scanner(System.in);
          try {
            int a = scanner.nextInt();
            int b = scanner.nextInt();
            int result = a / b;
            System.out.println(result);
        }
        catch (InputMismatchException e) {
            System.out.println("java.util.InputMismatchException");
        }
        catch (ArithmeticException e) {
            System.out.println("java.lang.ArithmeticException: / by zero");
        }
        finally {
            scanner.close();
        }
    }
}
```

372)
SHRIMA



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

```
10
3
```

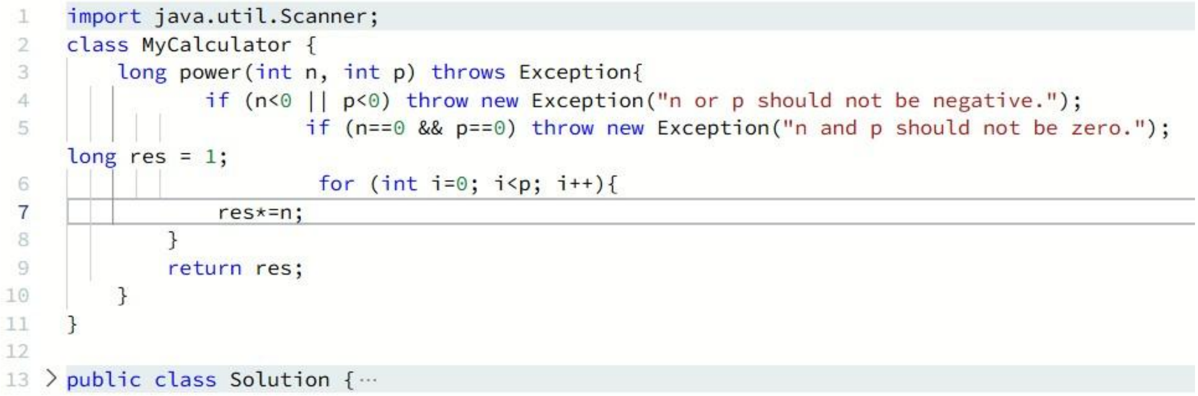
Your Output (stdout)

```
3
```

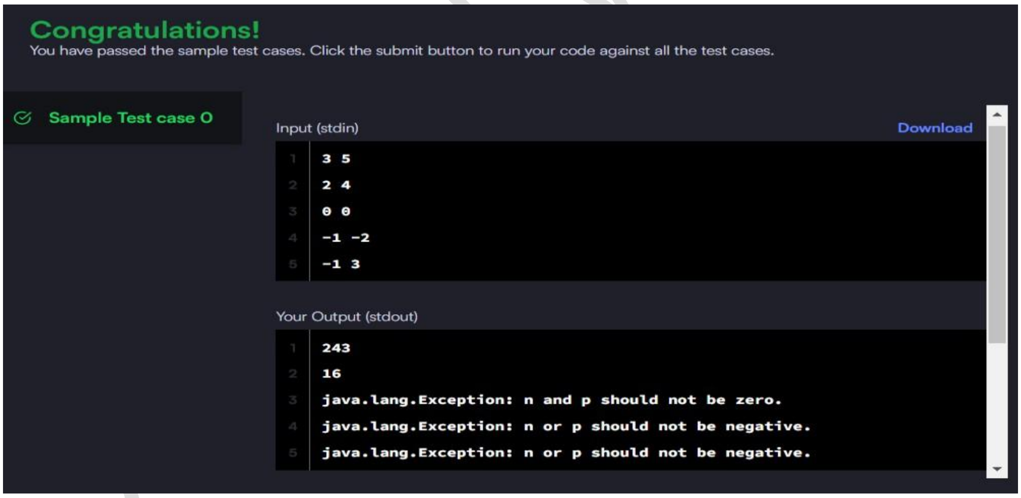
Expected Output

```
3
```

b)



```java
import java.util.Scanner;
class MyCalculator {
    long power(int n, int p) throws Exception{
           if (n<0 || p<0) throw new Exception("n or p should not be negative.");
                   if (n==0 && p==0) throw new Exception("n and p should not be zero.");
long res = 1;
                    for (int i=0; i<p; i++){
            res*=n;
        }
        return res;
    }
}

public class Solution {⋯
```



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

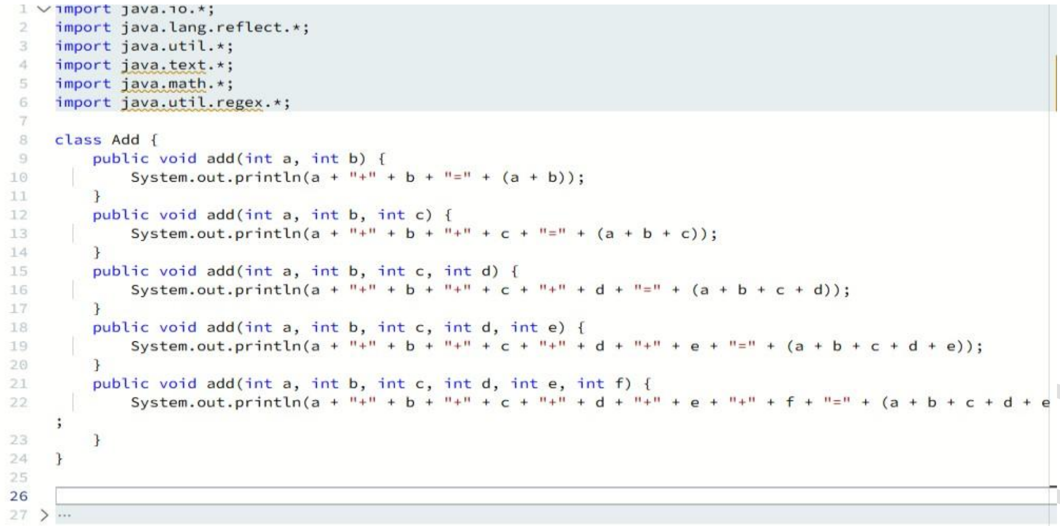
```
3 5
2 4
0 0
-1 -2
-1 3
```

Your Output (stdout)

```
243
16
java.lang.Exception: n and p should not be zero.
java.lang.Exception: n or p should not be negative.
java.lang.Exception: n or p should not be negative.
```

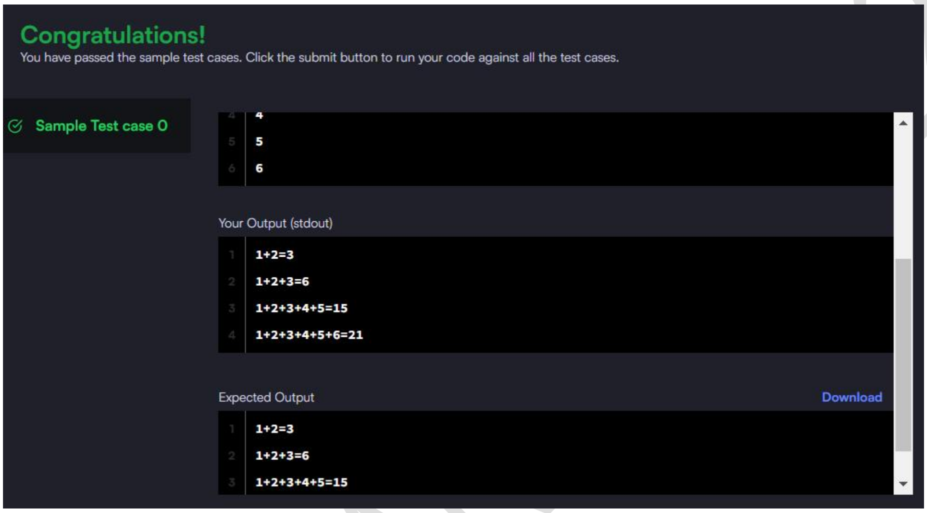
1B372)

c)



```
import java.io.*;
import java.lang.reflect.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

class Add {
    public void add(int a, int b) {
        System.out.println(a + "+" + b + "=" + (a + b));
    }
    public void add(int a, int b, int c) {
        System.out.println(a + "+" + b + "+" + c + "=" + (a + b + c));
    }
    public void add(int a, int b, int c, int d) {
        System.out.println(a + "+" + b + "+" + c + "+" + d + "=" + (a + b + c + d));
    }
    public void add(int a, int b, int c, int d, int e) {
        System.out.println(a + "+" + b + "+" + c + "+" + d + "+" + e + "=" + (a + b + c + d + e));
    }
    public void add(int a, int b, int c, int d, int e, int f) {
        System.out.println(a + "+" + b + "+" + c + "+" + d + "+" + e + "+" + f + "=" + (a + b + c + d + e
;
    }
}


> ...
```

OUTPUT c)



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

```
4
5
6
```

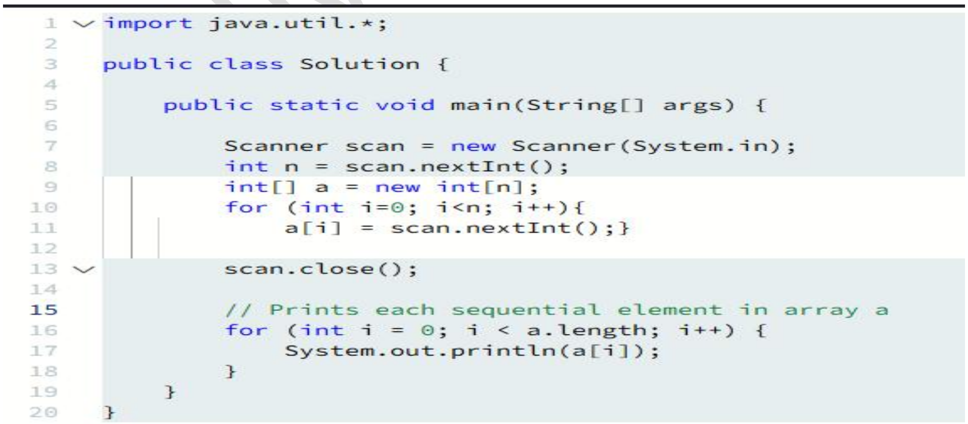
Your Output (stdout)

```
1+2=3
1+2+3=6
1+2+3+4+5=15
1+2+3+4+5+6=21
```

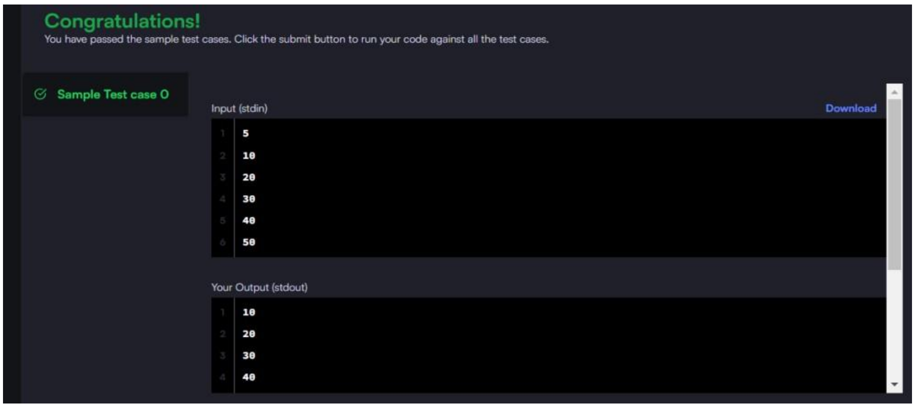
Expected Output

```
1+2=3
1+2+3=6
1+2+3+4+5=15
```

d)



```
import java.util.*;

public class Solution {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        int n = scan.nextInt();
        int[] a = new int[n];
        for (int i=0; i<n; i++){
            a[i] = scan.nextInt();}

        scan.close();

        // Prints each sequential element in array a
        for (int i = 0; i < a.length; i++) {
            System.out.println(a[i]);
        }
    }
}
```



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

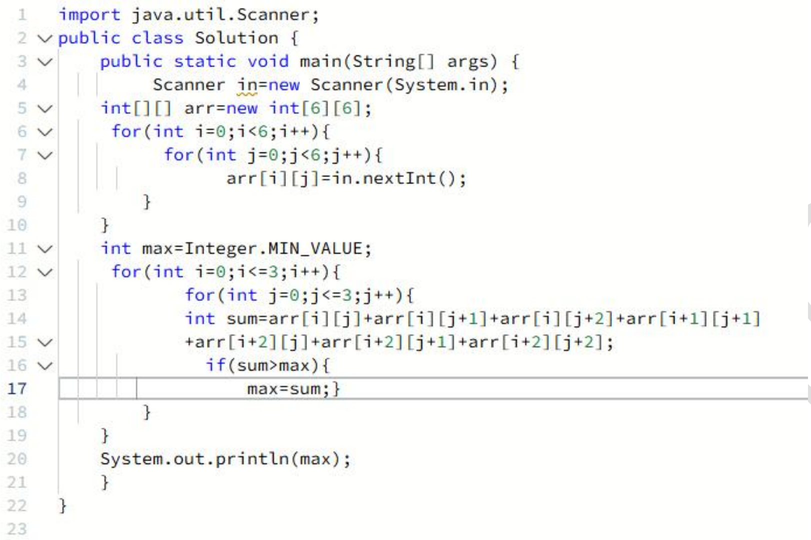
Input (stdin)

```
5
10
20
30
40
50
```

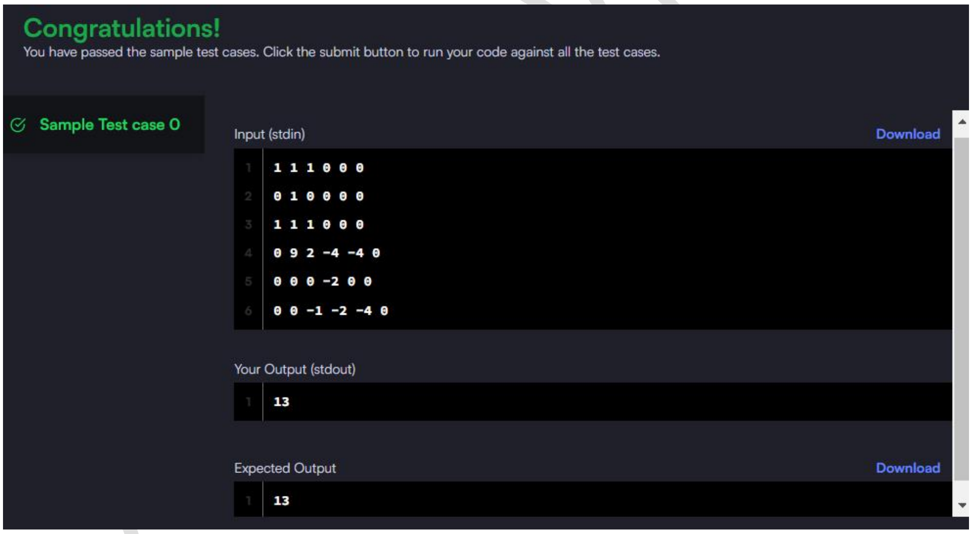
Your Output (stdout)

```
10
20
30
40
```

e)



```java
import java.util.Scanner;
public class Solution {
    public static void main(String[] args) {
        Scanner in=new Scanner(System.in);
    int[][] arr=new int[6][6];
     for(int i=0;i<6;i++){
          for(int j=0;j<6;j++){
                arr[i][j]=in.nextInt();
         }
     }
    int max=Integer.MIN_VALUE;
     for(int i=0;i<=3;i++){
            for(int j=0;j<=3;j++){
            int sum=arr[i][j]+arr[i][j+1]+arr[i][j+2]+arr[i+1][j+1]
            +arr[i+2][j]+arr[i+2][j+1]+arr[i+2][j+2];
              if(sum>max){
                  max=sum;}
         }
     }
    System.out.println(max);
    }
}
```



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

```
1 1 1 0 0 0
0 1 0 0 0 0
1 1 1 0 0 0
0 9 2 -4 -4 0
0 0 0 -2 0 0
0 0 -1 -2 -4 0
```

Your Output (stdout)

```
13
```

Expected Output

```
13
```

(221B372)

Lab Exercise 5

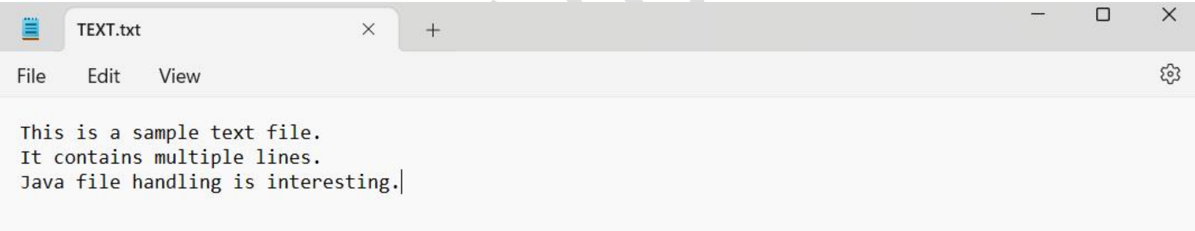
1. Write a Java program to create a text file named "TEXT.txt" on the disk. Add some text to this file. Enhance the program to read the file and display the following information on the screen in two columns: a. Number of lines b. Number of words c. Number of characters

```
  import java.io.FileWriter;

  import java.io.FileReader;

  import java.io.BufferedReader;

  import java.io.File;

  import java.io.IOException;

  public class TextFileInfo {

    public static void main(String[] args)

      {String filename = "TEXT.txt";

try (FileWriter writer = new FileWriter(filename))

{writer.write("This is a sample text file.\n");

        writer.write("It contains multiple lines.\n");

        writer.write("Java file handling is interesting.\n");

        System.out.println("Text written to file successfully.");

      } catch (IOException e) {

        System.out.println("An error occurred while writing to the file.");

        e.printStackTrace();

      }

      int lineCount = 0;

      int wordCount = 0;

      int charCount = 0;

      try (BufferedReader reader = new BufferedReader(new FileReader(filename))) {
```

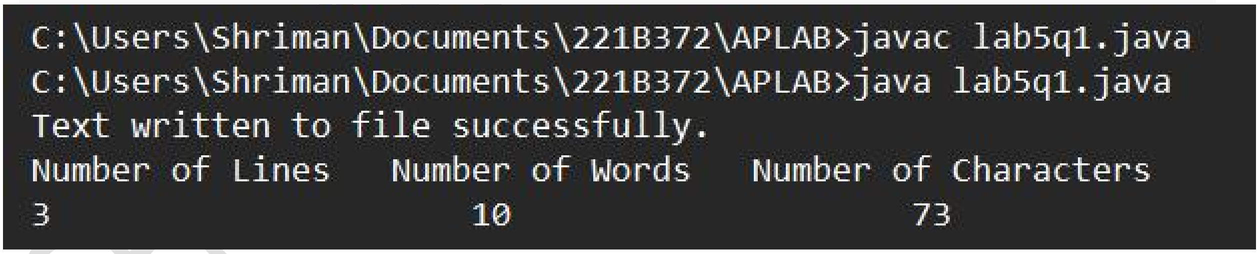
SHRIMAN VIJAY(221B372)

```
            String line;
            while ((line = reader.readLine()) != null)
               {lineCount++;
               wordCount += line.split("\\s+").length;
               charCount += line.length();
            }
         } catch (IOException e) {
            System.out.println("An error occurred while reading the file.");
            e.printStackTrace();
         }
         System.out.printf("%-20s %-20s %-20s%n", "Number of Lines", "Number of Words",
"Number of Characters");
         System.out.printf("%-20d %-20d %-20d%n", lineCount, wordCount, charCount);
      }
   }
```

OUTPUT



```
This is a sample text file.
It contains multiple lines.
Java file handling is interesting.
```



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q1.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q1.java
Text written to file successfully.
Number of Lines      Number of Words      Number of Characters
3                    10                   73
```

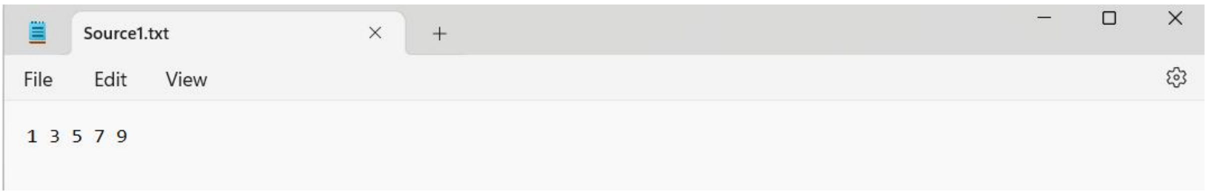
SHRIMAN VIJAY(221B372)

2. Create two files, 'Source1' and 'Source2', each containing a sorted list of integers. Develop a Java program to read the contents of both files, merge the lists into a sorted form, and save the result in a new file named 'Target'.

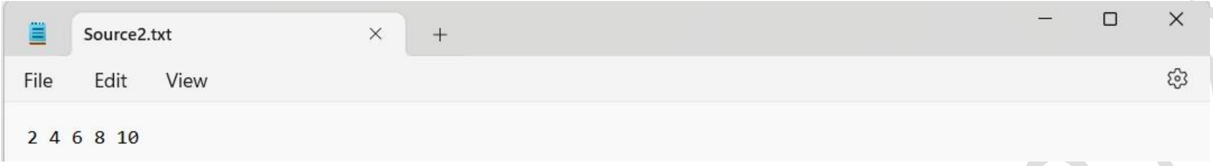
```
import java.io.*;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class lab5q2 {

   public static void main(String[] args) {
      try (FileWriter writer1 = new FileWriter("Source1.txt");
         FileWriter writer2 = new FileWriter("Source2.txt"))
         {writer1.write("1\n3\n5\n7\n9\n");
         writer1.close();
         writer2.write("2\n4\n6\n8\n10\n");
         writer2.close();

         System.out.println("Source1.txt and Source2.txt created with sorted integers.");
      } catch (IOException e) {
         System.out.println("An error occurred while creating source files.");
         e.printStackTrace();
      }
      List<Integer> numbers = new ArrayList<>();
      readAndAddNumbers("Source1.txt", numbers);
      readAndAddNumbers("Source2.txt", numbers);
      Collections.sort(numbers);
      try (FileWriter writer = new FileWriter("Target.txt"))
         {for (int number : numbers) {
            writer.write(number + "\n");
         }
         System.out.println("Merged sorted list written to Target.txt successfully.");
      } catch (IOException e) {
         System.out.println("An error occurred while writing to Target.txt.");
         e.printStackTrace();
      }
   }
   private static void readAndAddNumbers(String filename, List<Integer> numbers)
      {try (BufferedReader reader = new BufferedReader(new FileReader(filename)))
      {
         String line;
         while ((line = reader.readLine()) != null)
            { numbers.add(Integer.parseInt(line.trim()));
         }
      } catch (IOException e) {
         System.out.println("An error occurred while reading from " + filename);
         e.printStackTrace();
      }
   }
}
```

OUTPUT :

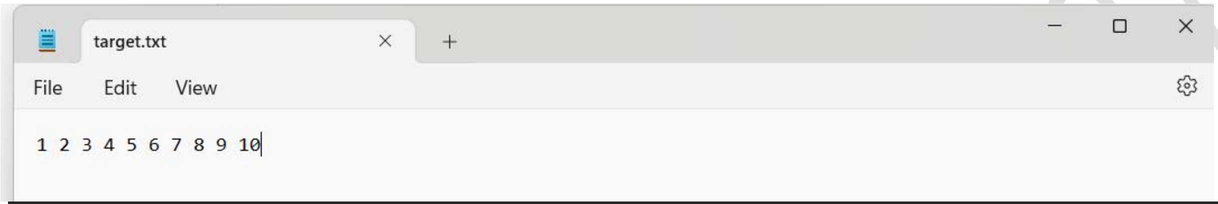
SHRIMAN VIJAY(221B372)



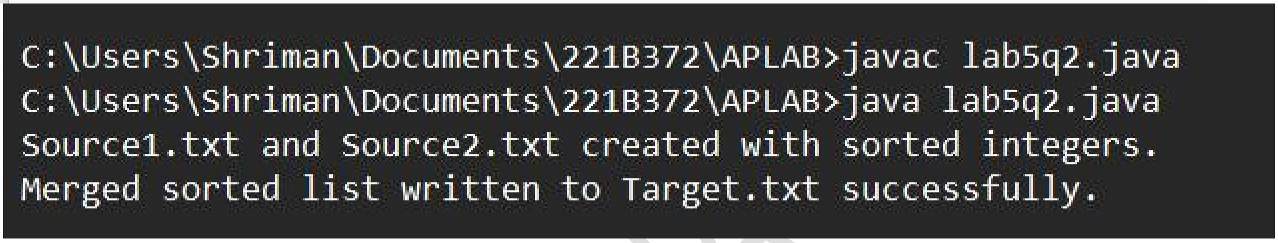
```
1 3 5 7 9
```



```
2 4 6 8 10
```



```
1 2 3 4 5 6 7 8 9 10
```



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q2.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q2.java
Source1.txt and Source2.txt created with sorted integers.
Merged sorted list written to Target.txt successfully.
```

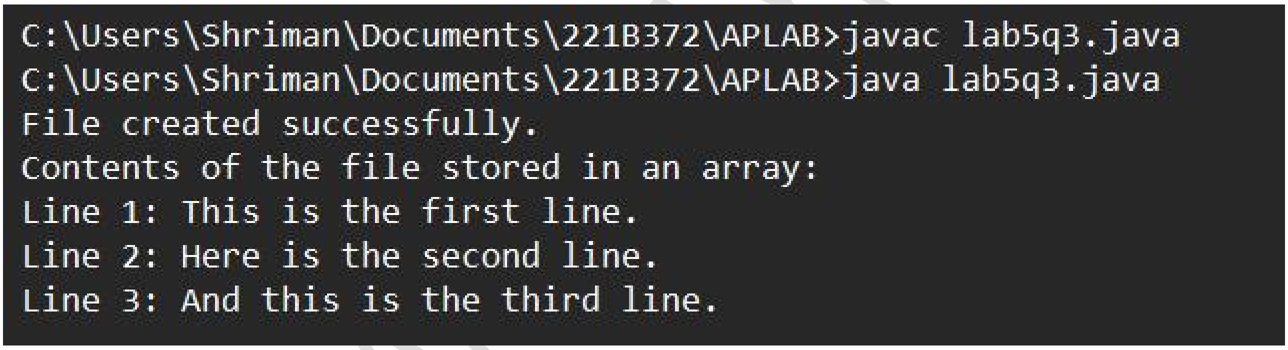
SHRIMAN VIJAY (221B372)

3. Write a program to read a file and store each line into an array.

```
import java.io.*;
import java.util.ArrayList;
import java.util.List;

public class lab5q3 {

  public static void main(String[] args)
    {String filename = "ExampleFile.txt";

    try (FileWriter writer = new FileWriter(filename))
      {writer.write("This is the first line.\n");
      writer.write("Here is the second line.\n");
      writer.write("And this is the third line.\n");
      System.out.println("File created successfully.");
    } catch (IOException e) {
      System.out.println("An error occurred while writing to the file.");
      e.printStackTrace();
```

```
        List<String> linesList = new ArrayList<>();

        try (BufferedReader reader = new BufferedReader(new FileReader(filename)))
            {String line;
            while ((line = reader.readLine()) != null)
                {linesList.add(line);
            }
        } catch (IOException e) {
            System.out.println("An error occurred while reading the file.");
            e.printStackTrace();
        }
        String[] linesArray = linesList.toArray(new String[0]);

        System.out.println("Contents of the file stored in an array:");
        for (int i = 0; i < linesArray.length; i++) {
            System.out.println("Line " + (i + 1) + ": " + linesArray[i]);
        }
    }
}
```

OUTPUT



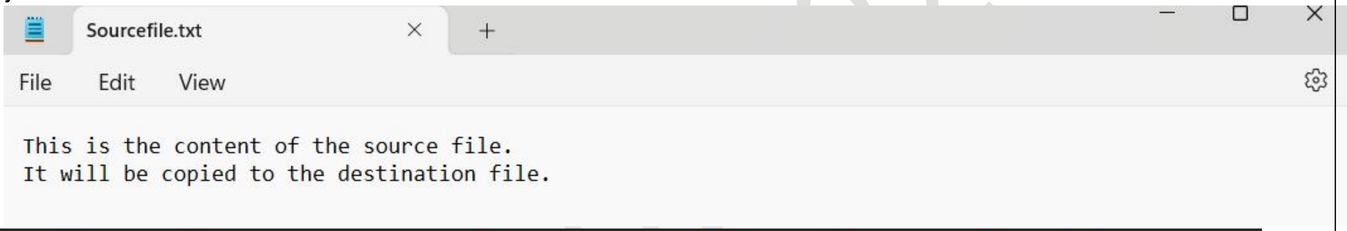
```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q3.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q3.java
File created successfully.
Contents of the file stored in an array:
Line 1: This is the first line.
Line 2: Here is the second line.
Line 3: And this is the third line.
```

4. Develop a program to copy the contents of one file to another

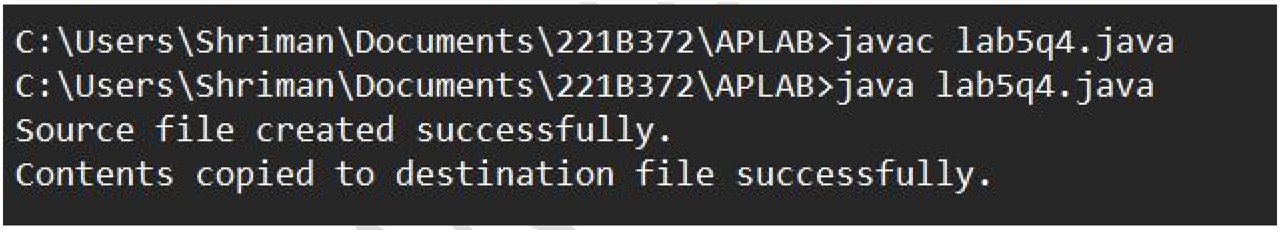
```
import java.io.*;
public class lab5q4 {
    public static void main(String[] args)
        {String sourceFile = "SourceFile.txt";
        String destinationFile = "DestinationFile.txt";

        try (FileWriter writer = new FileWriter(sourceFile)) {
            writer.write("This is the content of the source file.\n");
            writer.write("It will be copied to the destination file.\n");
            System.out.println("Source file created successfully.");
        } catch (IOException e) {
            System.out.println("An error occurred while creating the source file.");
            e.printStackTrace();
```

SHRIMAN VIJAY(221B372)

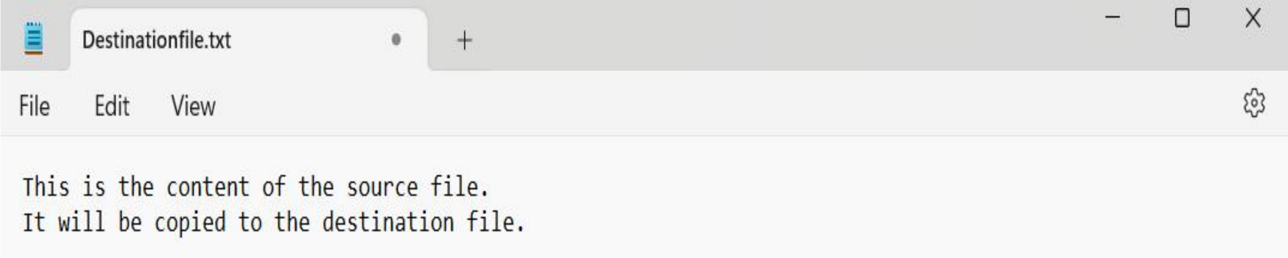
```
}

        try (BufferedReader reader = new BufferedReader(new FileReader(sourceFile));
             FileWriter writer = new FileWriter(destinationFile)) {

            String line;
            while ((line = reader.readLine()) != null)
               {writer.write(line + "\n");
            }
            System.out.println("Contents copied to destination file successfully.");
        } catch (IOException e) {
            System.out.println("An error occurred while copying the file.");
            e.printStackTrace();
        }
    }
}
```



```
This is the content of the source file.
It will be copied to the destination file.
```



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q4.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q4.java
Source file created successfully.
Contents copied to destination file successfully.
```



```
This is the content of the source file.
It will be copied to the destination file.
```

5. Implement a program to merge the contents of two files and write the result to a new file.

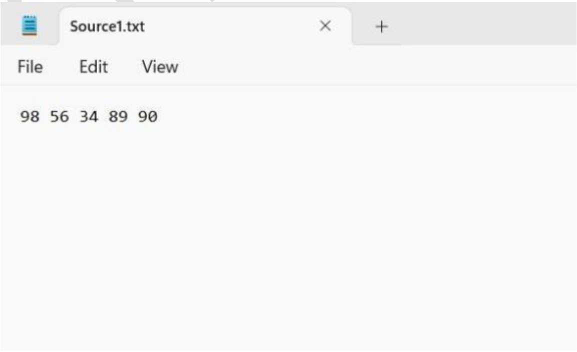
```
import java.io.*;
        public class lab5q5 {
           public static void main(String[] args)
               { String sourceFileName1 =
```

Shriman Vijay (221B372)

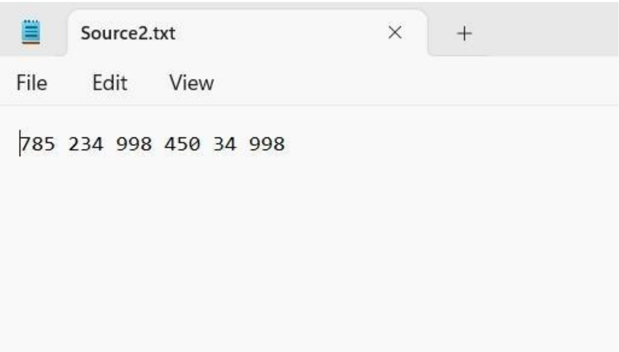
```
        "source1.txt";
        String sourceFileName2 = "source2.txt";

       String targetFileName = "merged.txt";

try (BufferedReader reader1 = new
BufferedReader(newFileReader(sourceFileName1));

BufferedReader reader2 = new BufferedReader(new
FileReader(sourceFileName2));

  BufferedWriter writer = new BufferedWriter(new
FileWriter(targetFileName))) {

            String line;

while ((line reader1.readLine()) != null)

  {writer.write(line);

writer.newLine();

            }

while ((line = reader2.readLine()) != null)

{writer.write(line);

writer.newLine();

            }

            System.out.println("Files merged successfully!");

        }

 catch (IOException e) {

            System.out.println("An error occurred during file
merging.");            e.printStackTrace();

}}}
```

SHRIMAN VIJAY(221B372)

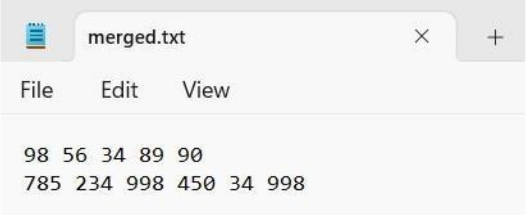
OUTPUT:



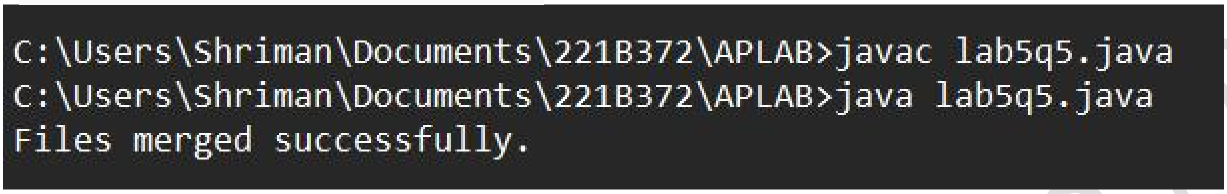
```
98 56 34 89 90
```



```
785 234 998 450 34 998
```



```
98 56 34 89 90
785 234 998 450 34 998
```



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q5.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q5.java
Files merged successfully.
```

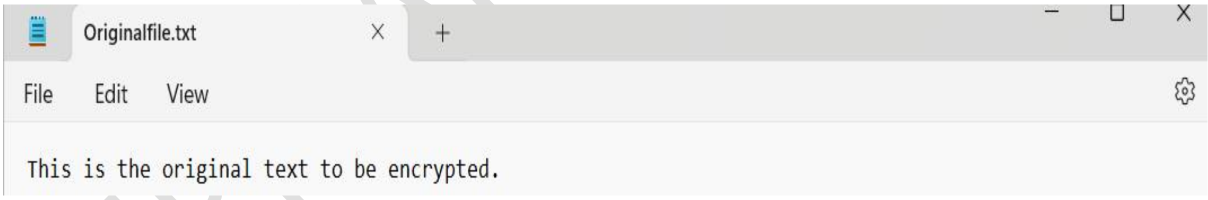
6.Write a program to encrypt and decrypt a text file.

```
import java.io.*;
public class lab5q6 {
  private static final int SHIFT = 3;
  public static void main(String[] args) {
    String originalFile = "OriginalFile.txt";
    String encryptedFile = "EncryptedFile.txt";
    String decryptedFile = "DecryptedFile.txt";
    try (FileWriter writer = new FileWriter(originalFile))
      { writer.write("This is the original text to be
      encrypted.");System.out.println("Original file created
      successfully.");
    } catch (IOException e)
      {e.printStackTrace();
    }
    encryptFile(originalFile, encryptedFile);
    System.out.println("File encrypted successfully.");

    decryptFile(encryptedFile, decryptedFile);
    System.out.println("File decrypted successfully.");
  }
  public static void encryptFile(String inputFile, String outputFile) {
```

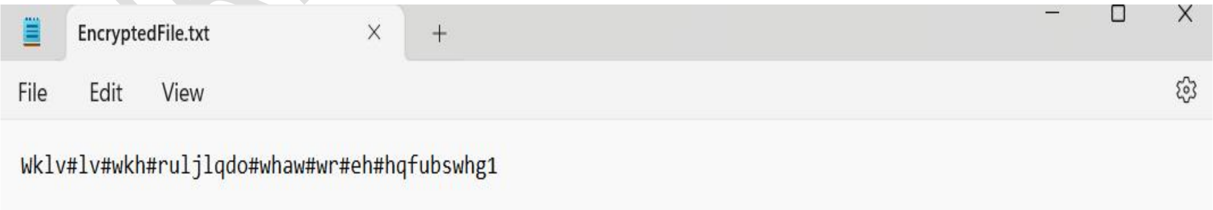
SHRIMAN VIJAY(221B372)

```
        try (BufferedReader reader = new BufferedReader(new FileReader(inputFile):

        FileWriter writer = new FileWriter(outputFile)) {int character;
            while ((character = reader.read()) != -1)

                {writer.write(character + SHIFT);

            }

        } catch (IOException e)

            {e.printStackTrace();

        }

    }
    public static void decryptFile(String inputFile, String outputFile) {

        try (BufferedReader reader = new BufferedReader(new FileReader(inputFile));

             FileWriter writer = new FileWriter(outputFile)) {

            int character;

            while ((character = reader.read()) != -1)

                {writer.write(character - SHIFT);

            }

        } catch (IOException e)

            {e.printStackTrace();

        }}}
```

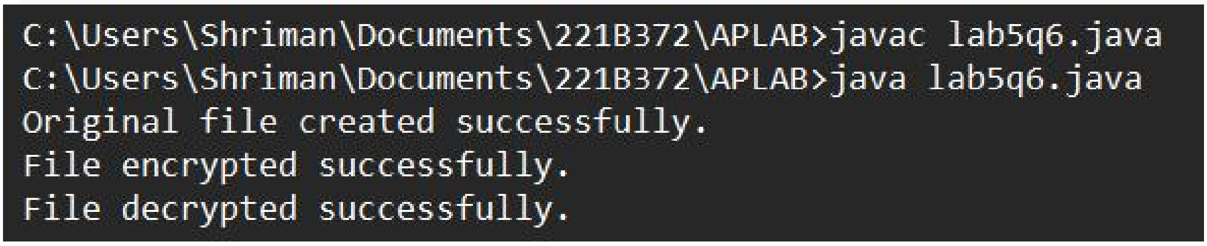
OUPUT :



This is the original text to be encrypted.



Wklv#lv#wkh#ruljlqdo#whaw#wr#eh#hqfubswhg1



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q6.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q6.java
Original file created successfully.
File encrypted successfully.
File decrypted successfully.
```

AN VIJAY(221B372)

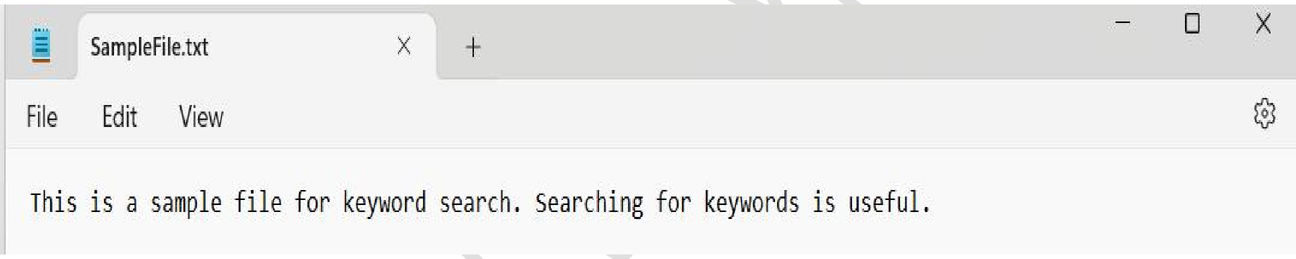
7.Create a program that allows the user to search for a specific keyword in a single file and display its index from the start of file.

```
import java.io.*;

import java.util.Scanner;

public class lab5q7{

   public static void main(String[] args)

      {String filename = "SampleFile.txt";

      String keyword;

      try (FileWriter writer = new FileWriter(filename)) {

         writer.write("This is a sample file for keyword search. Searching for keywords is
useful.");

         System.out.println("File created successfully.");

      } catch (IOException e)

         {e.printStackTrace();

      }

      Scanner scanner = new Scanner(System.in);

      System.out.print("Enter the keyword to search: ");

      keyword = scanner.nextLine();

      searchKeywordInFile(filename, keyword);

   }

   public static void searchKeywordInFile(String filename, String keyword) {

      try (BufferedReader reader = new BufferedReader(new FileReader(filename)))

         {String line;

         int index = 0;

         boolean found = false;

         while ((line = reader.readLine()) != null) {

            int keywordIndex = line.indexOf(keyword);

            if (keywordIndex != -1) {
```

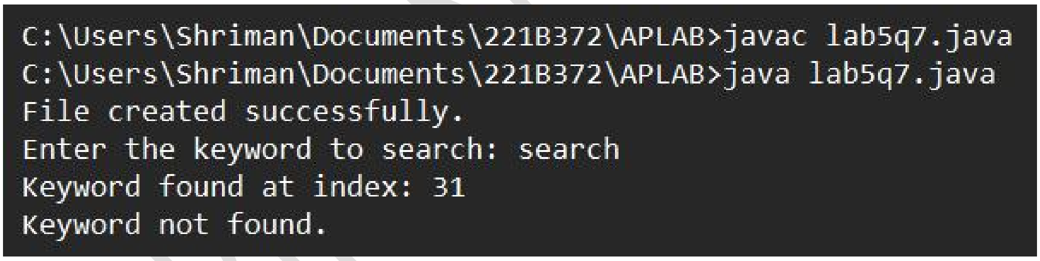
SHRIMAN VIJAY(221B372)

```
                System.out.println("Keyword found at index: " + (index + keywordIndex));
                found = true;
                break;
            }
            index += line.length() + 1;
        }
        if (!found) {
            System.out.println("Keyword not found.");
        }
    } catch (IOException e)
        {e.printStackTrace();
}}}
```

OUTPUT:



This is a sample file for keyword search. Searching for keywords is useful.



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab5q7.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab5q7.java
File created successfully.
Enter the keyword to search: search
Keyword found at index: 31
Keyword not found.
```

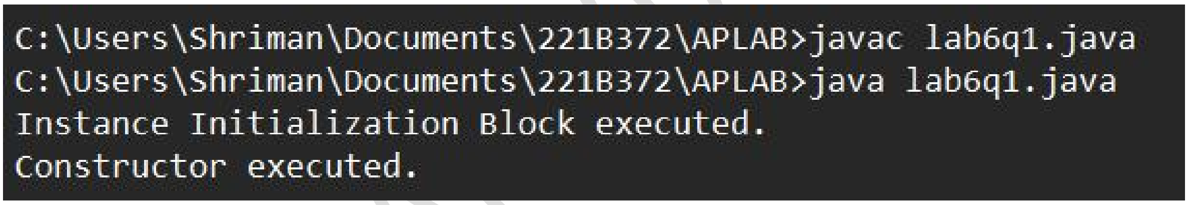
SHRIMAN VIJAY(221B372)

Lab Exercise 6

1. Write a Java program to demonstrate that the instance initialization block will be executed before the constructor of a class.

```
public class lab6q1 {
  {
    System.out.println("Instance Initialization Block executed.");
  }
  public InitializationDemo() {
    System.out.println("Constructor executed.");
  }
  public static void main(String[] args) {
    InitializationDemo obj = new InitializationDemo();
  }
}
```

OUTPUT:



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab6q1.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab6q1.java
Instance Initialization Block executed.
Constructor executed.
```

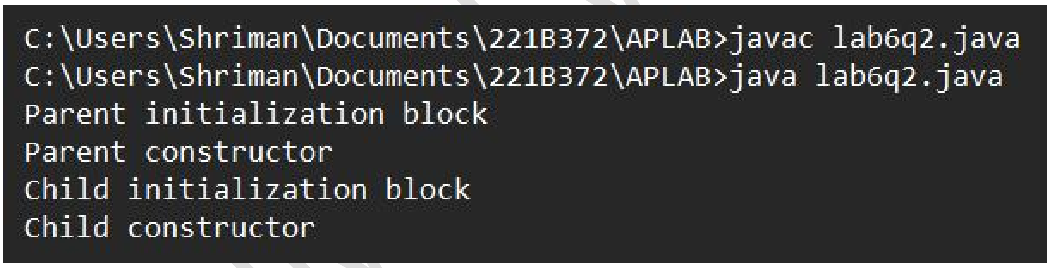
2. Write a Java program to test the execution order of the initialization blocks in both the parent and child classes when the constructor of the parent class is explicitly called using super

```
class Parent {
  {
    System.out.println("Parent initialization block");
  }
  public Parent() {
    System.out.println("Parent constructor");
```

SHRIMAN VIJAY(221B372)

```
  }
}
class Child extends Parent {
  {
    System.out.println("Child initialization block");
  }
  public Child()
    {super();
    System.out.println("Child constructor");
  }
}
public class lab6q2{
  public static void main(String[] args)
    {new Child();
  }
}
```

OUTPUT:



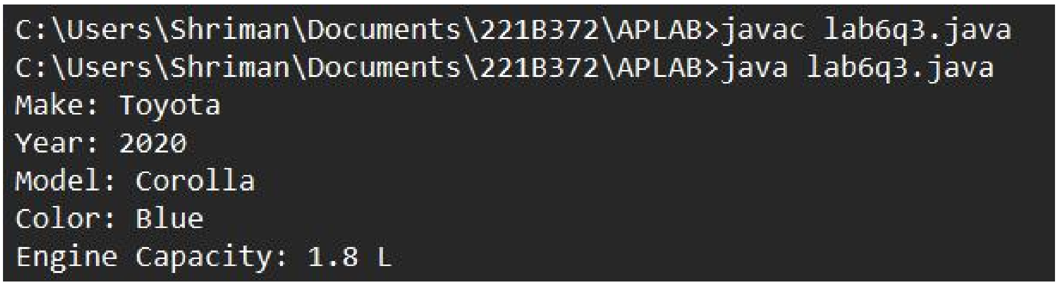
```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab6q2.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab6q2.java
Parent initialization block
Parent constructor
Child initialization block
Child constructor
```

3. You are required to design a Java program to manage a vehicle system using inheritance, demonstrating constructor and method execution across parent and child classes. ● Parent Class: Vehicle ○ Instance Variables: ■ make(Manufacturer of the vehicle) ■ year(Year of manufacture) ○ Constructor: Initializes the two instance variables (make and year). ○ Method: displayVehicleInfo() that prints the values of make and year. ● ChildClass: Car ○ Instance Variables: ■ model(Car model) ■ color (Color of the car) ■ engineCapacity (Engine capacity of the car in liters) ○ Constructor: Initializes the three instance variables (model, color, engineCapacity) and calls the parent class constructor to initialize the inherited variables. ○ Method: displayCarInfo() that displays the values of all instance variables (including the ones inherited from the Vehicle class). In the main() method, create an instance of Car by passing values for both Vehicle and Car class variables, and then display all details using the methods provided.

SHRIMAN VIJAY(221B372)

```
class    Vehicle
   {        String
   make;      int
   year;
   public Vehicle(String make, int year)
      {this.make = make;
      this.year = year;
   }
   public void displayVehicleInfo()
      { System.out.println("Make: " +
      make);System.out.println("Year: " +
      year);
   }
}
class Car extends Vehicle
   {String model;
   String color;
   double engineCapacity;

   public Car(String make, int year, String model, String color, double engineCapacity)
      {super(make, year);
      this.model = model;
      this.color = color;
      this.engineCapacity = engineCapacity;
   }

   public void displayCarInfo()
      { displayVehicleInfo();
      System.out.println("Model: " + model);
      System.out.println("Color: " + color);
      System.out.println("Engine Capacity: " + engineCapacity + " L");
   }
}
public class lab6q3 {
   public static void main(String[] args) {
      Car car = new Car("Toyota", 2020, "Corolla", "Blue", 1.8);
      car.displayCarInfo();
   }
}
```

OUTPUT:



```
C:\Users\Shriman\Documents\221B372\APLAB>javac lab6q3.java
C:\Users\Shriman\Documents\221B372\APLAB>java lab6q3.java
Make: Toyota
Year: 2020
Model: Corolla
Color: Blue
Engine Capacity: 1.8 L
```

SHRIMAN VIJAY(221B372)

4. Solve the following questions:

a. https://www.hackerrank.com/challenges/java-substring/problem?isFullScreen=true

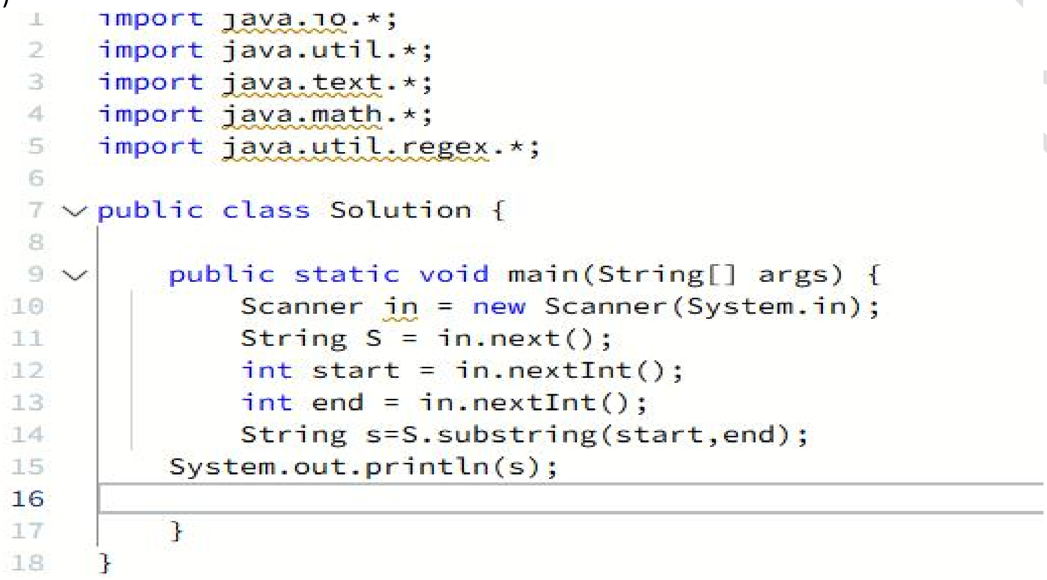
b. https://www.hackerrank.com/challenges/java-string-compare/problem?isFullScreen=true

c. https://www.hackerrank.com/challenges/java-string-reverse/problem?isFullScreen=true

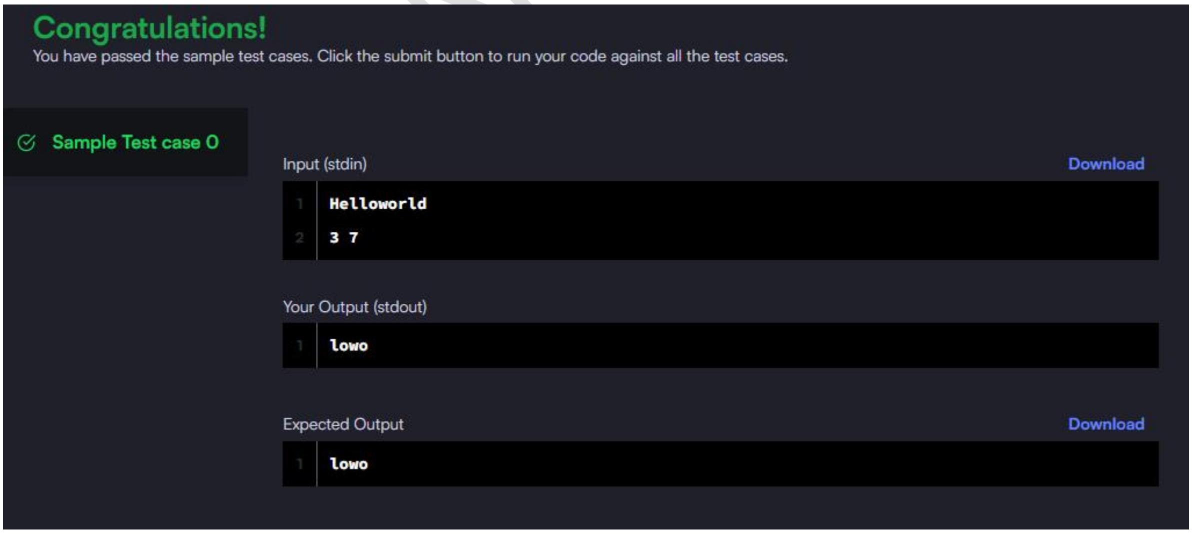
d. https://www.hackerrank.com/challenges/java-string-tokens/problem?isFullScreen=true

e. https://www.hackerrank.com/challenges/java-end-of-file/problem?isFullScreen=true

a)



```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        String S = in.next();
        int start = in.nextInt();
        int end = in.nextInt();
        String s=S.substring(start,end);
    System.out.println(s);

    }
}
```



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

```
Helloworld
3 7
```

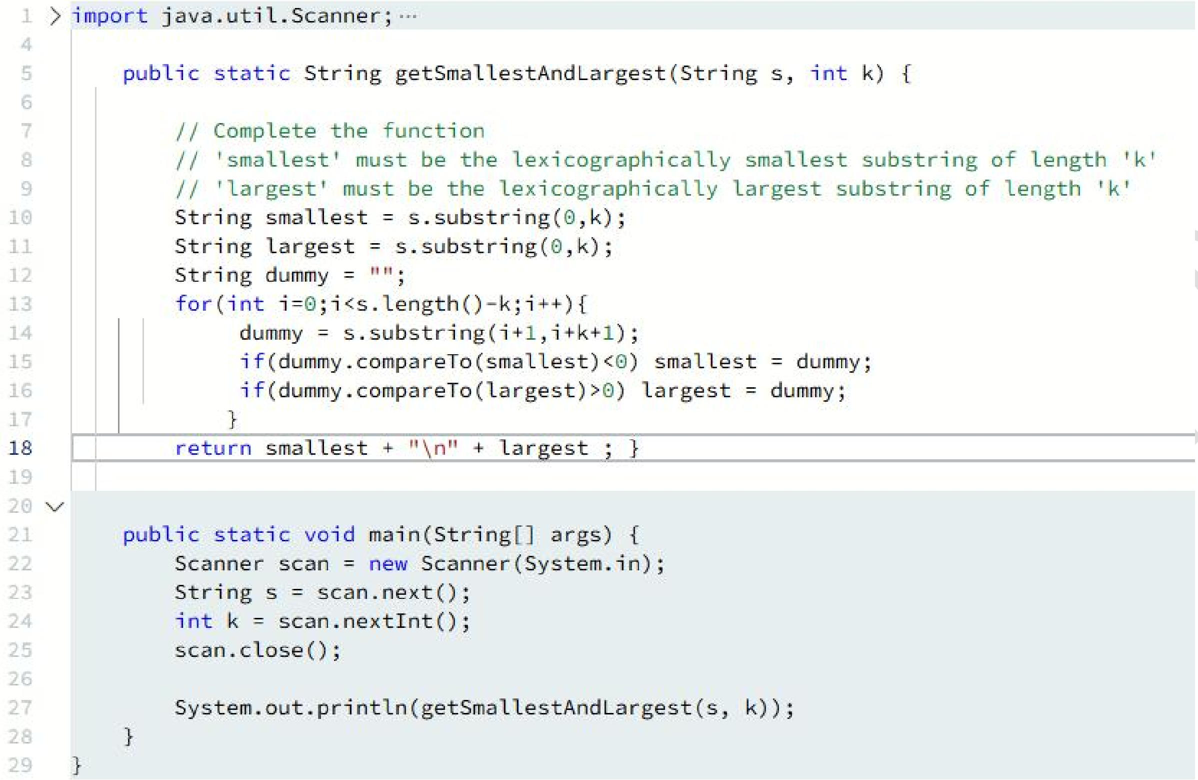
Your Output (stdout)

```
lowo
```

Expected Output

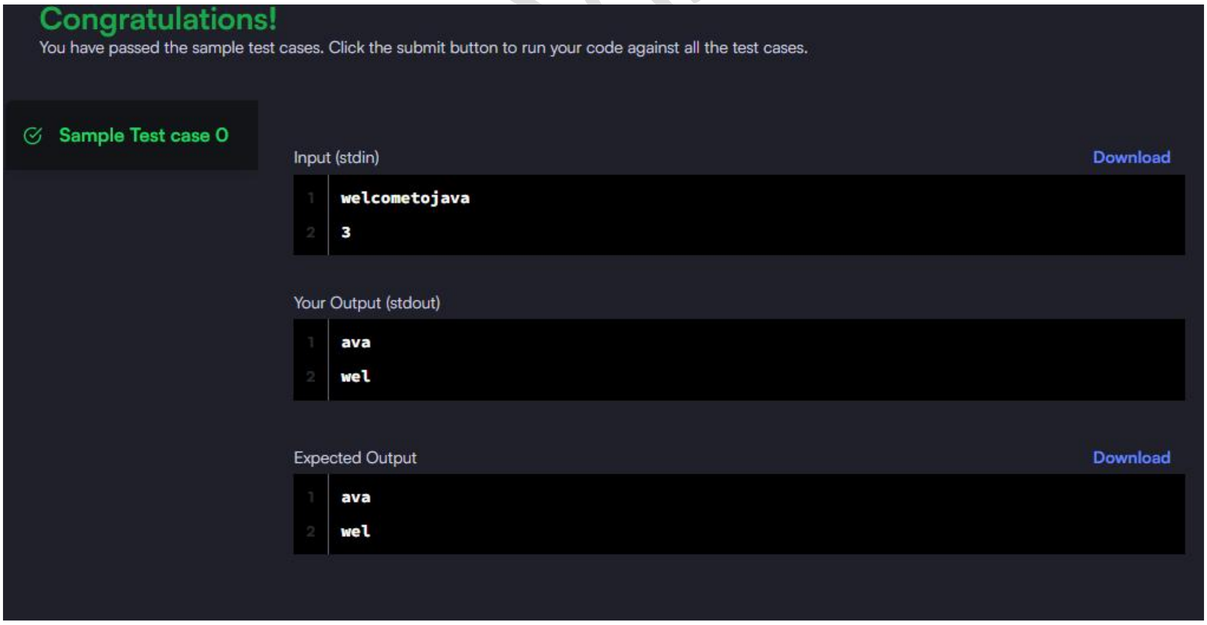
```
lowo
```

b)



```java
import java.util.Scanner;···

    public static String getSmallestAndLargest(String s, int k) {

        // Complete the function
        // 'smallest' must be the lexicographically smallest substring of length 'k'
        // 'largest' must be the lexicographically largest substring of length 'k'
        String smallest = s.substring(0,k);
        String largest = s.substring(0,k);
        String dummy = "";
        for(int i=0;i<s.length()-k;i++){
             dummy = s.substring(i+1,i+k+1);
             if(dummy.compareTo(smallest)<0) smallest = dummy;
             if(dummy.compareTo(largest)>0) largest = dummy;
            }
        return smallest + "\n" + largest ; }


    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        String s = scan.next();
        int k = scan.nextInt();
        scan.close();

        System.out.println(getSmallestAndLargest(s, k));
    }
}
```

OUTPUT:



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

```
welcometojava
3
```

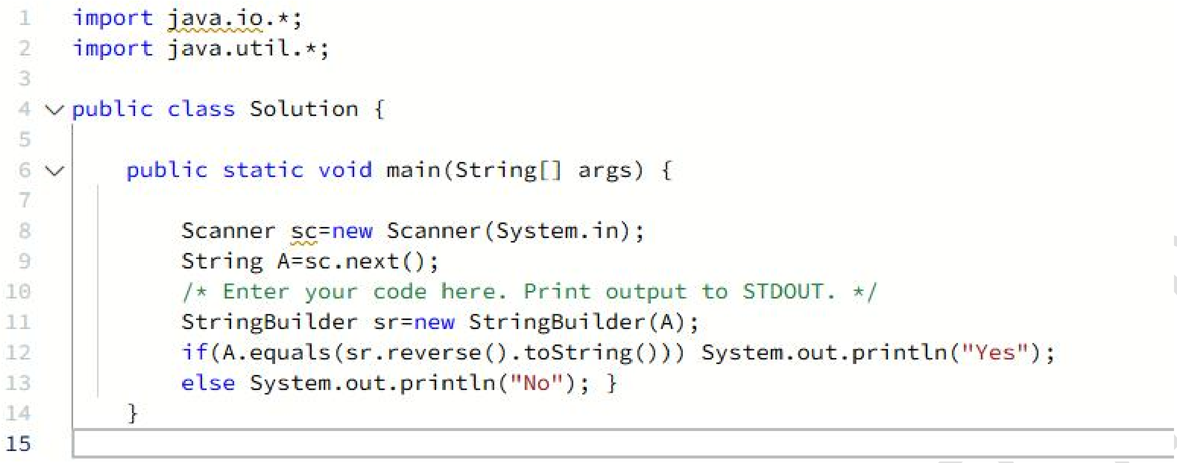
Your Output (stdout)

```
ava
wel
```

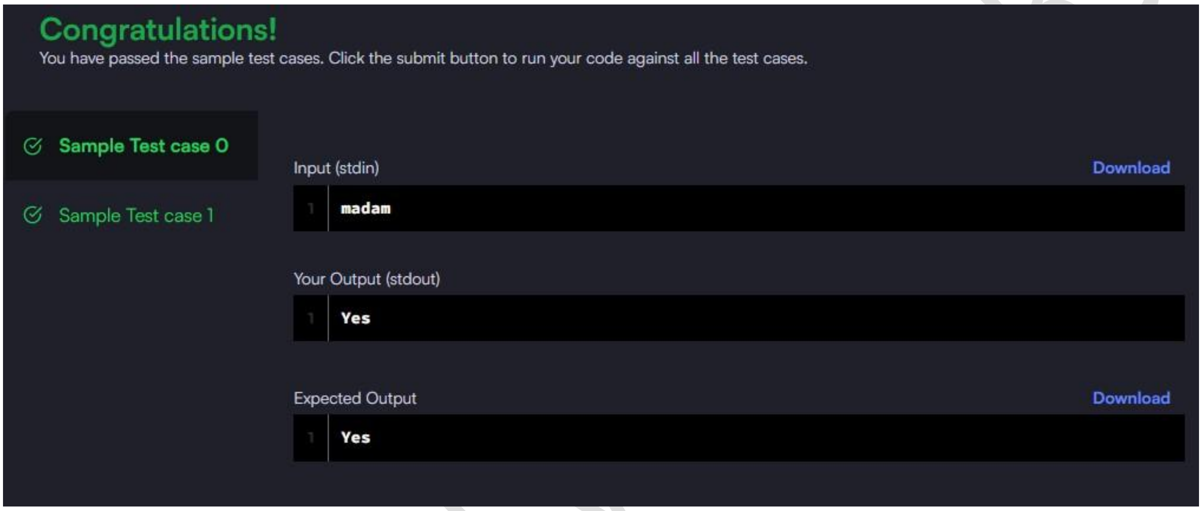
Expected Output

```
ava
wel
```

c)



```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {

        Scanner sc=new Scanner(System.in);
        String A=sc.next();
        /* Enter your code here. Print output to STDOUT. */
        StringBuilder sr=new StringBuilder(A);
        if(A.equals(sr.reverse().toString())) System.out.println("Yes");
        else System.out.println("No"); }
    }

```



Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Input (stdin)

```
madam
```

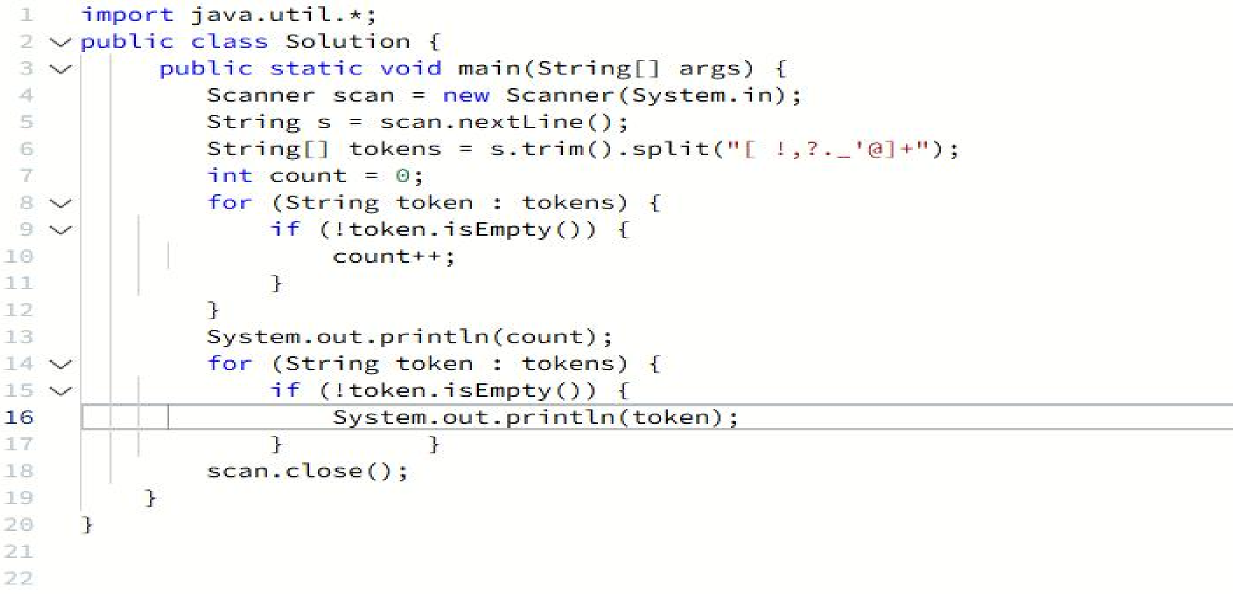
Your Output (stdout)

```
Yes
```

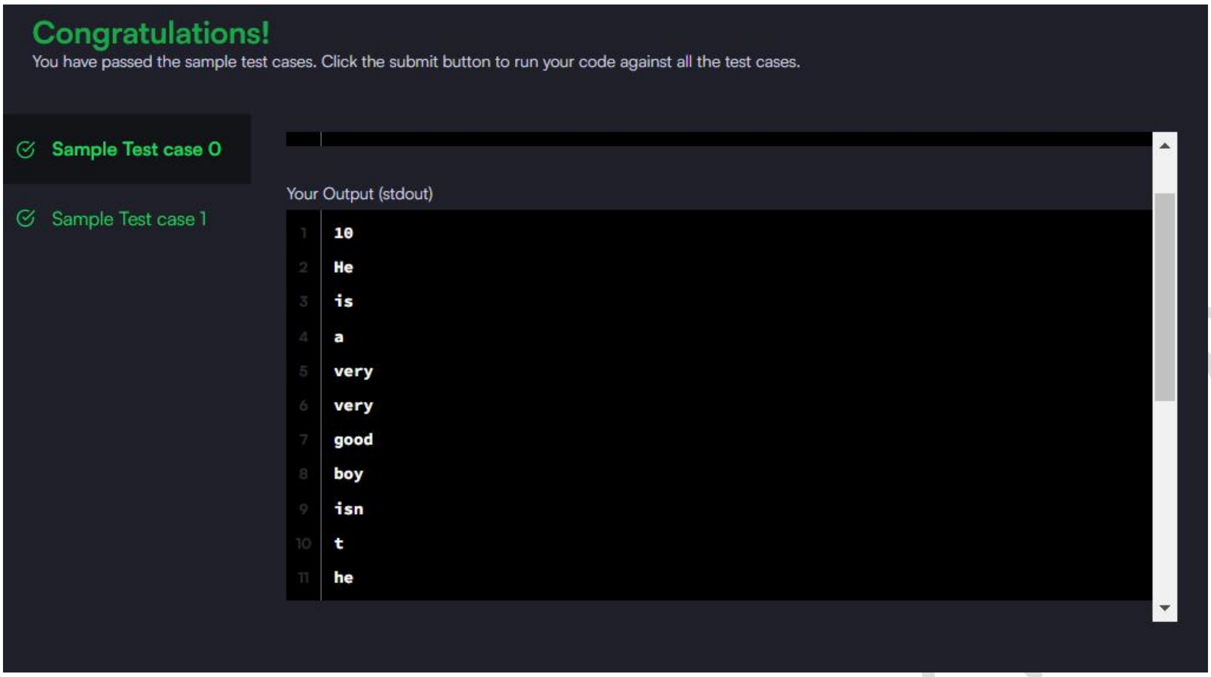
Expected Output

```
Yes
```

d)



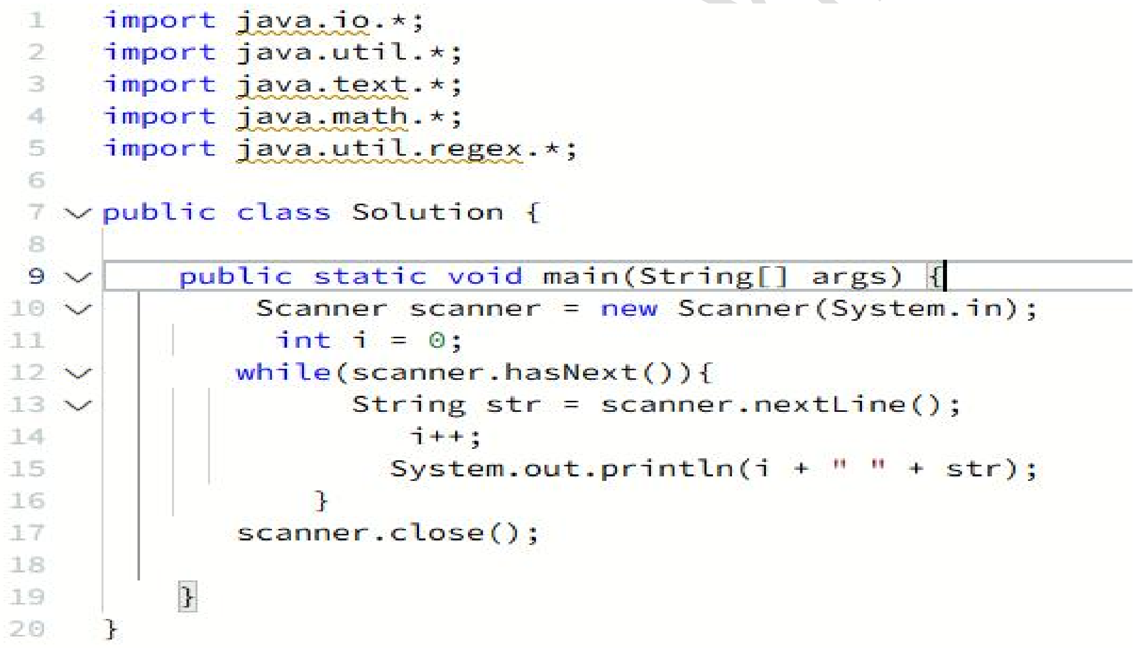
```java
import java.util.*;
public class Solution {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        String s = scan.nextLine();
        String[] tokens = s.trim().split("[ !,?._'@]+");
        int count = 0;
        for (String token : tokens) {
            if (!token.isEmpty()) {
                count++;
            }
        }
        System.out.println(count);
        for (String token : tokens) {
            if (!token.isEmpty()) {
                System.out.println(token);
            }          }
        scan.close();
    }
}


```



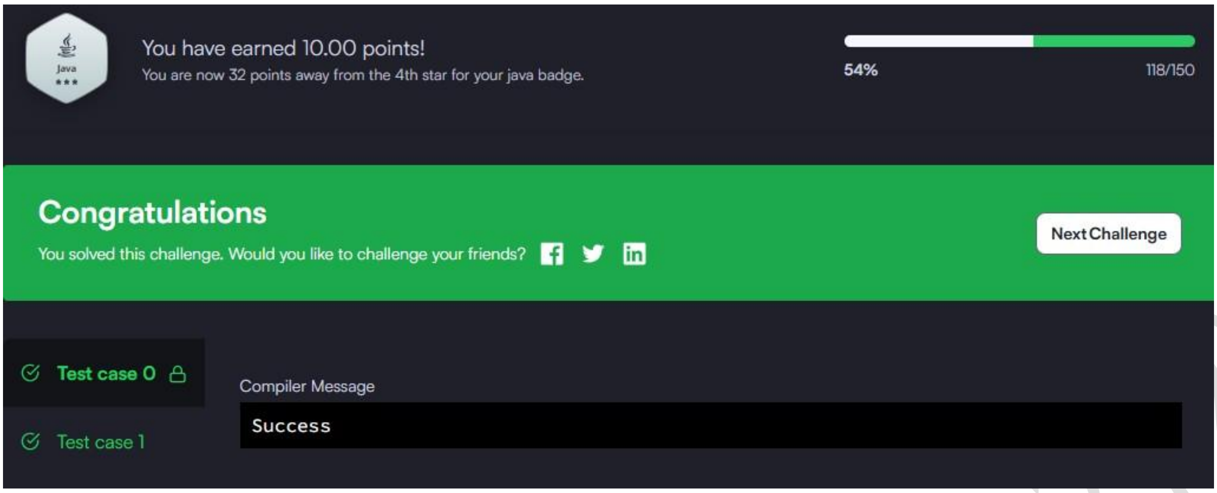
Congratulations!
You have passed the sample test cases. Click the submit button to run your code against all the test cases.
Sample Test case 0
Sample Test case 1
Your Output (stdout)

```
10
He
is
a
very
very
good
boy
isn
t
he
```

e)



```java
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
         Scanner scanner = new Scanner(System.in);
          int i = 0;
       while(scanner.hasNext()){
             String str = scanner.nextLine();
                  i++;
               System.out.println(i + " " + str);
          }
       scanner.close();

    }
}
```



You have earned 10.00 points!

You are now 32 points away from the 4th star for your java badge.

Congratulations

You solved this challenge. Would you like to challenge your friends?

Compiler Message

```
Success
```

SHRIMAN VIJAY(221B372)