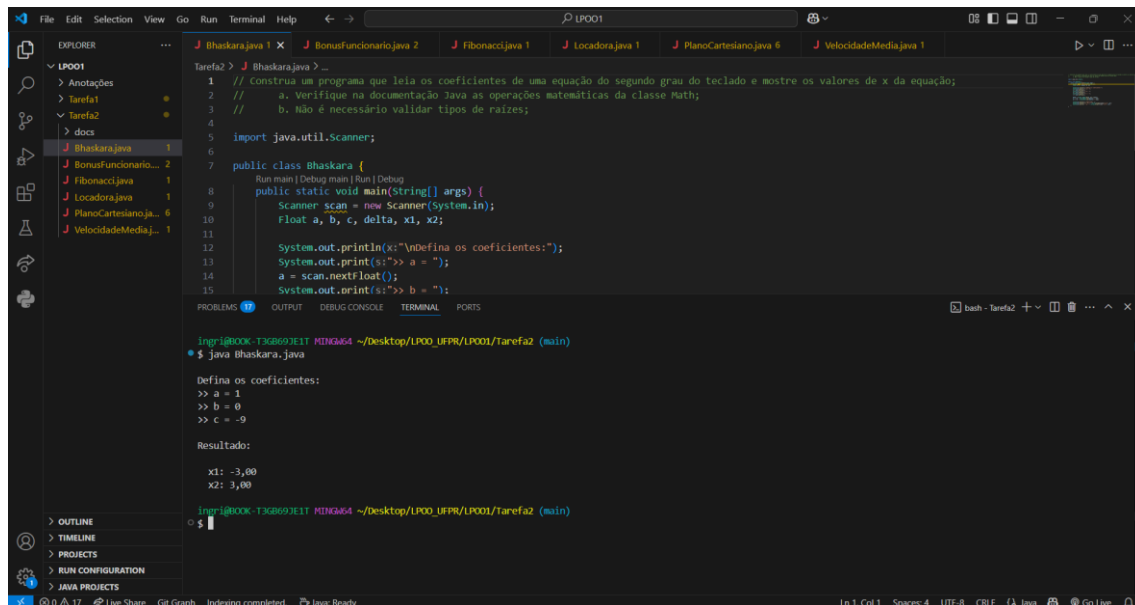


Nome: Ingrid Cristine Rocha

GRR20242220

Evidências – Tarefa 2

Bhaskara.java



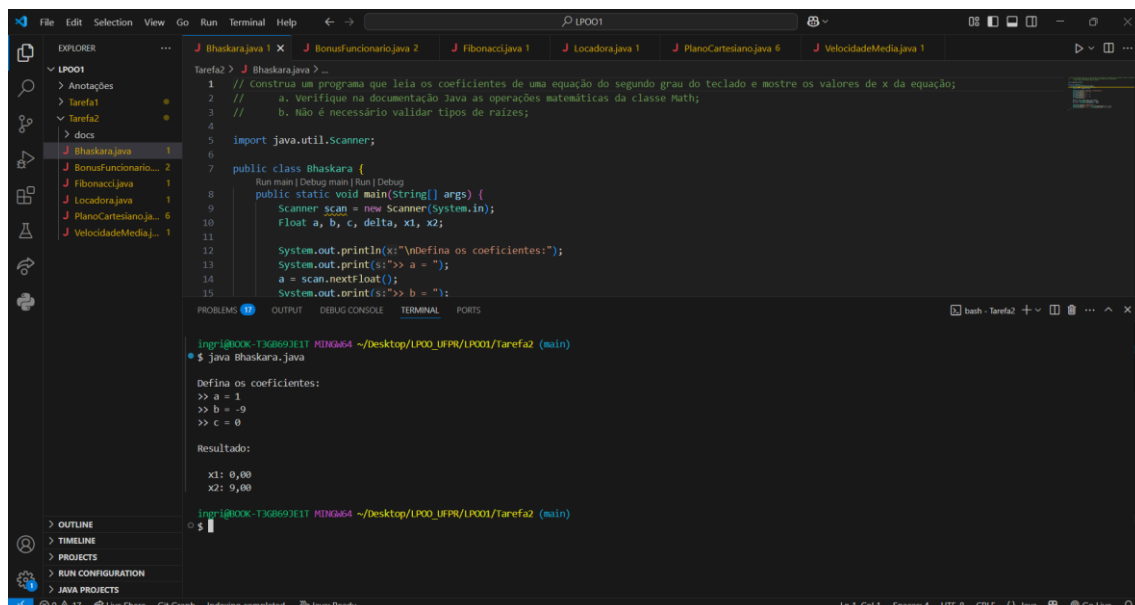
The screenshot shows an IDE with the file explorer on the left displaying a project named 'LPOO1'. The file 'Bhaskara.java' is selected. The main editor shows the code for the 'Bhaskara' class, which reads coefficients 'a', 'b', and 'c' from the user and calculates the roots 'x1' and 'x2' of a quadratic equation. The terminal at the bottom shows the command 'java Bhaskara.java' being executed, followed by the input of coefficients 'a=1', 'b=0', and 'c=-9', and the resulting roots 'x1: -3,00' and 'x2: 3,00'.

```
1 // Construa um programa que leia os coeficientes de uma equação do segundo grau do teclado e mostre os valores de x da equação;
2 // a. Verifique na documentação Java as operações matemáticas da classe Math;
3 // b. Não é necessário validar tipos de raízes;
4
5 import java.util.Scanner;
6
7 public class Bhaskara {
8     Run main | Debug main | Run | Debug
9     public static void main(String[] args) {
10         Scanner scan = new Scanner(System.in);
11         float a, b, c, delta, x1, x2;
12
13         System.out.println(x: "Defina os coeficientes:");
14         System.out.print(s: ">> a = ");
15         a = scan.nextFloat();
16         System.out.print(s: ">> b = ");
17         b = scan.nextFloat();
18         System.out.print(s: ">> c = ");
19         c = scan.nextFloat();
20
21         delta = b*b - 4*a*c;
22         x1 = (-b + Math.sqrt(delta)) / (2*a);
23         x2 = (-b - Math.sqrt(delta)) / (2*a);
24
25         System.out.println("Resultado:");
26         System.out.println("x1: " + x1);
27         System.out.println("x2: " + x2);
28     }
29 }
```

bash - Tarefa2

```
ingrid@BOOK-T3GB693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$ java Bhaskara.java
Defina os coeficientes:
>> a = 1
>> b = 0
>> c = -9
Resultado:
x1: -3,00
x2: 3,00
ingrid@BOOK-T3GB693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$
```

Teste 1



This screenshot is identical to the one above, showing the same code and terminal output for Teste 1. The terminal output shows the same input and results.

```
1 // Construa um programa que leia os coeficientes de uma equação do segundo grau do teclado e mostre os valores de x da equação;
2 // a. Verifique na documentação Java as operações matemáticas da classe Math;
3 // b. Não é necessário validar tipos de raízes;
4
5 import java.util.Scanner;
6
7 public class Bhaskara {
8     Run main | Debug main | Run | Debug
9     public static void main(String[] args) {
10         Scanner scan = new Scanner(System.in);
11         float a, b, c, delta, x1, x2;
12
13         System.out.println(x: "Defina os coeficientes:");
14         System.out.print(s: ">> a = ");
15         a = scan.nextFloat();
16         System.out.print(s: ">> b = ");
17         b = scan.nextFloat();
18         System.out.print(s: ">> c = ");
19         c = scan.nextFloat();
20
21         delta = b*b - 4*a*c;
22         x1 = (-b + Math.sqrt(delta)) / (2*a);
23         x2 = (-b - Math.sqrt(delta)) / (2*a);
24
25         System.out.println("Resultado:");
26         System.out.println("x1: " + x1);
27         System.out.println("x2: " + x2);
28     }
29 }
```

bash - Tarefa2

```
ingrid@BOOK-T3GB693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$ java Bhaskara.java
Defina os coeficientes:
>> a = 1
>> b = 0
>> c = 0
Resultado:
x1: 0,00
x2: 0,00
ingrid@BOOK-T3GB693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$
```

Teste 2

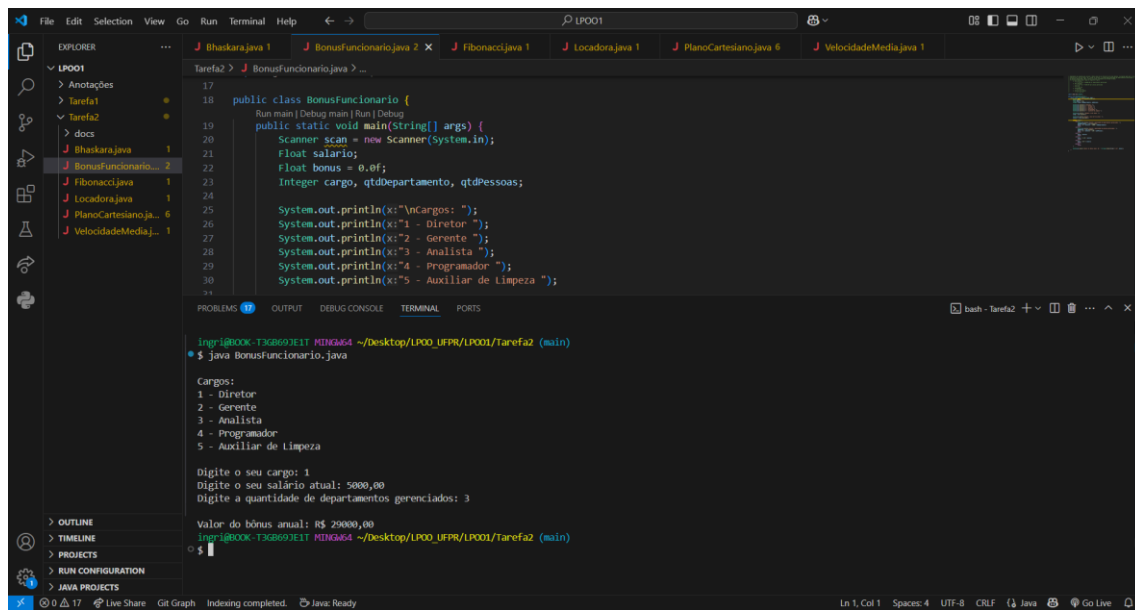
```
File Edit Selection View Go Run Terminal Help
LPOO1
Tarefa2 > J Bhaskara.java 1
1 // Construa um programa que leia os coeficientes de uma equação do segundo grau do teclado e mostre os valores de x da equação;
2 // a. Verifique na documentação Java as operações matemáticas da classe Math;
3 // b. Não é necessário validar tipos de raízes;
4
5 import java.util.Scanner;
6
7 public class Bhaskara {
8     Run main | Debug main | Run | Debug
9     public static void main(String[] args) {
10         Scanner scan = new Scanner(System.in);
11         Float a, b, c, delta, x1, x2;
12
13         System.out.println(x1:"\nDefina os coeficientes:");
14         System.out.print(s1:"> a = ");
15         a = scan.nextFloat();
16         System.out.print(s1:"> b = ");
17         b = scan.nextFloat();
18         System.out.print(s1:"> c = ");
19         c = scan.nextFloat();
20         delta = 4*a*c - 4*b*b;
21         x1 = (-b + Math.sqrt(delta)) / (2*a);
22         x2 = (-b - Math.sqrt(delta)) / (2*a);
23         System.out.println("x1: " + x1 + " x2: " + x2);
24     }
25 }
26
27 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash - Tarefa2
Ingr1@BOOK-T3G8693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$ java Bhaskara.java
Defina os coeficientes:
>> a = 3
>> b = -7
>> c = 2
Resultado:
x1: 0,33
x2: 2,00
Ingr1@BOOK-T3G8693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$
```

Teste 3

```
File Edit Selection View Go Run Terminal Help
LPOO1
Tarefa2 > J Bhaskara.java 1
1 // Construa um programa que leia os coeficientes de uma equação do segundo grau do teclado e mostre os valores de x da equação;
2 // a. Verifique na documentação Java as operações matemáticas da classe Math;
3 // b. Não é necessário validar tipos de raízes;
4
5 import java.util.Scanner;
6
7 public class Bhaskara {
8     Run main | Debug main | Run | Debug
9     public static void main(String[] args) {
10         Scanner scan = new Scanner(System.in);
11         Float a, b, c, delta, x1, x2;
12
13         System.out.println(x1:"\nDefina os coeficientes:");
14         System.out.print(s1:"> a = ");
15         a = scan.nextFloat();
16         System.out.print(s1:"> b = ");
17         b = scan.nextFloat();
18         System.out.print(s1:"> c = ");
19         c = scan.nextFloat();
20         delta = 4*a*c - 4*b*b;
21         x1 = (-b + Math.sqrt(delta)) / (2*a);
22         x2 = (-b - Math.sqrt(delta)) / (2*a);
23         System.out.println("x1: " + x1 + " x2: " + x2);
24     }
25 }
26
27 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash - Tarefa2
Ingr1@BOOK-T3G8693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$ java Bhaskara.java
Defina os coeficientes:
>> a = -1
>> b = 4
>> c = -4
Resultado:
x1: 2,00
x2: 2,00
Ingr1@BOOK-T3G8693E1T MINGW64 ~/Desktop/LPOO_UFPR/LPOO1/Tarefa2 (main)
$
```

Teste 4

BonusFuncionario.java



The screenshot shows an IDE with the file `BonusFuncionario.java` open. The code defines a `BonusFuncionario` class with a `main` method that prompts the user for cargo, salary, and the number of departments, then calculates and displays the annual bonus.

```
17 public class BonusFuncionario {
18     Run main | Debug main | Run | Debug
19     public static void main(String[] args) {
20         Scanner scan = new Scanner(System.in);
21         Float salario;
22         Float bonus = 0.0f;
23         Integer cargo, qtdDepartamento, qtdPessoas;
24
25         System.out.println(x:"nCargos: ");
26         System.out.println(x:"1 - Diretor ");
27         System.out.println(x:"2 - Gerente ");
28         System.out.println(x:"3 - Analista ");
29         System.out.println(x:"4 - Programador ");
30         System.out.println(x:"5 - Auxiliar de Limpeza ");
31     }
```

The terminal output shows the execution of the program:

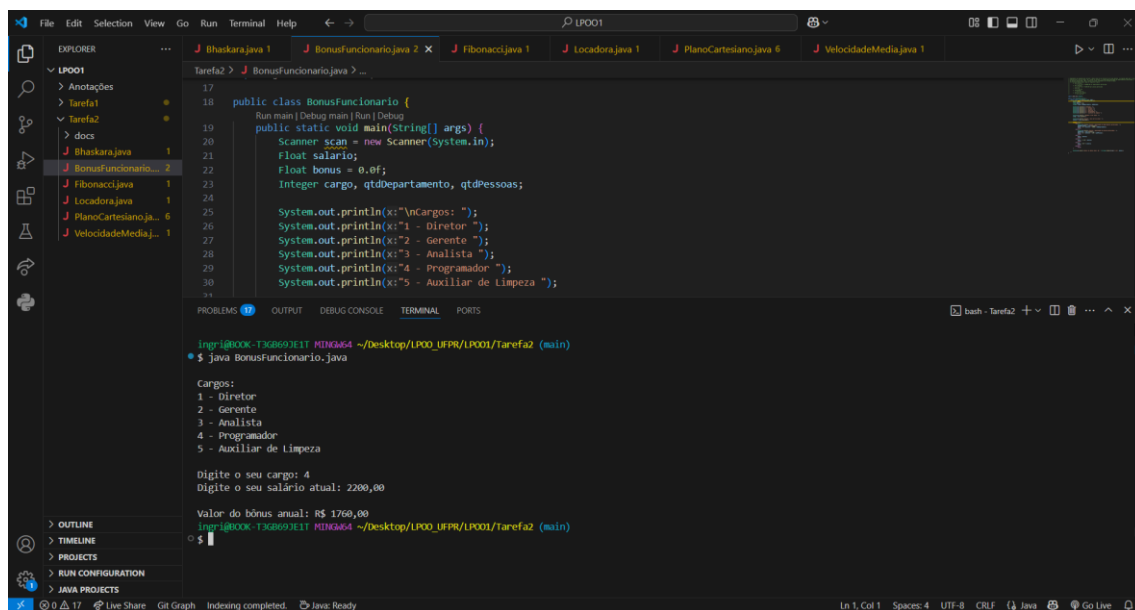
```
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java BonusFuncionario.java

Cargos:
1 - Diretor
2 - Gerente
3 - Analista
4 - Programador
5 - Auxiliar de Limpeza

Digite o seu cargo: 1
Digite o seu salário atual: 5000,00
Digite a quantidade de departamentos gerenciados: 3

Valor do bônus anual: R$ 20000,00
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste 1



The screenshot shows the same IDE with the `BonusFuncionario.java` file. The terminal output shows the execution of the program with different input values:

```
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java BonusFuncionario.java

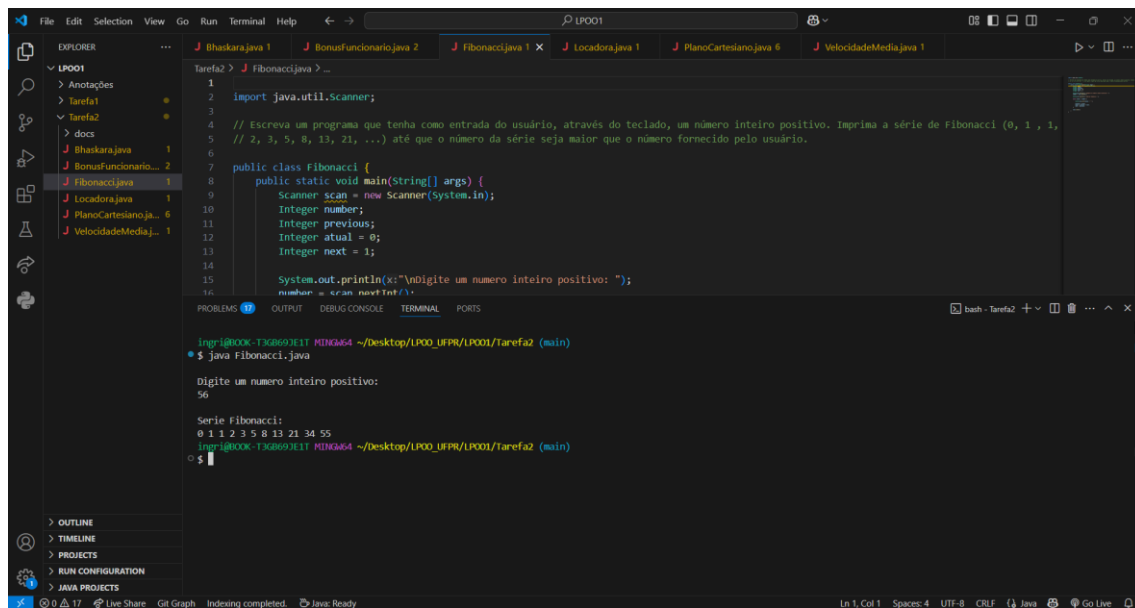
Cargos:
1 - Diretor
2 - Gerente
3 - Analista
4 - Programador
5 - Auxiliar de Limpeza

Digite o seu cargo: 4
Digite o seu salário atual: 2200,00

Valor do bônus anual: R$ 1760,00
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste 2

Fibonacci.java



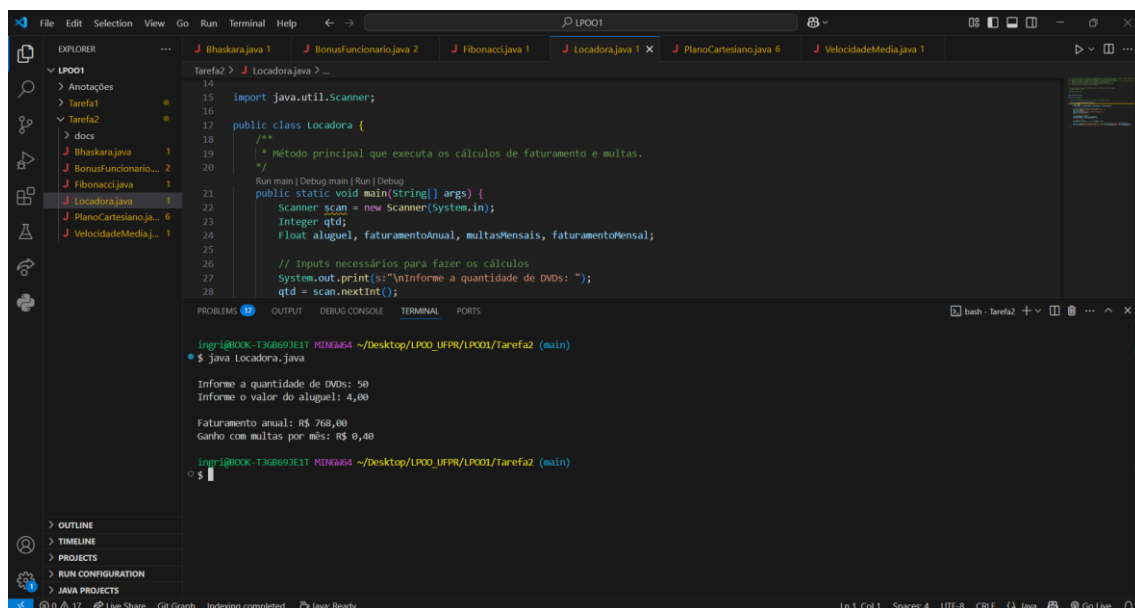
The screenshot shows an IDE with the file explorer on the left listing several Java files. The main editor displays the code for `Fibonacci.java`. The code imports `java.util.Scanner` and contains a `main` method that prompts the user for a positive integer. It then calculates and prints the Fibonacci sequence up to that number. The terminal at the bottom shows the command `java Fibonacci.java` being executed, with the user inputting `56` and the program outputting the sequence: `0 1 1 2 3 5 8 13 21 34 55`.

```
1 import java.util.Scanner;
2
3 // Escreva um programa que tenha como entrada do usuário, através do teclado, um número inteiro positivo. Imprima a série de Fibonacci (0, 1, 1,
4 // 2, 3, 5, 8, 13, 21, ...) até que o número da série seja maior que o número fornecido pelo usuário.
5
6 public class Fibonacci {
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9         Integer number;
10        Integer previous;
11        Integer atual = 0;
12        Integer next = 1;
13
14        System.out.println("Digite um numero inteiro positivo: ");
15        number = scan.nextInt();
16    }
17 }
```

```
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java Fibonacci.java
Digite um numero inteiro positivo:
56
Serie Fibonacci:
0 1 1 2 3 5 8 13 21 34 55
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste

Locadora.java



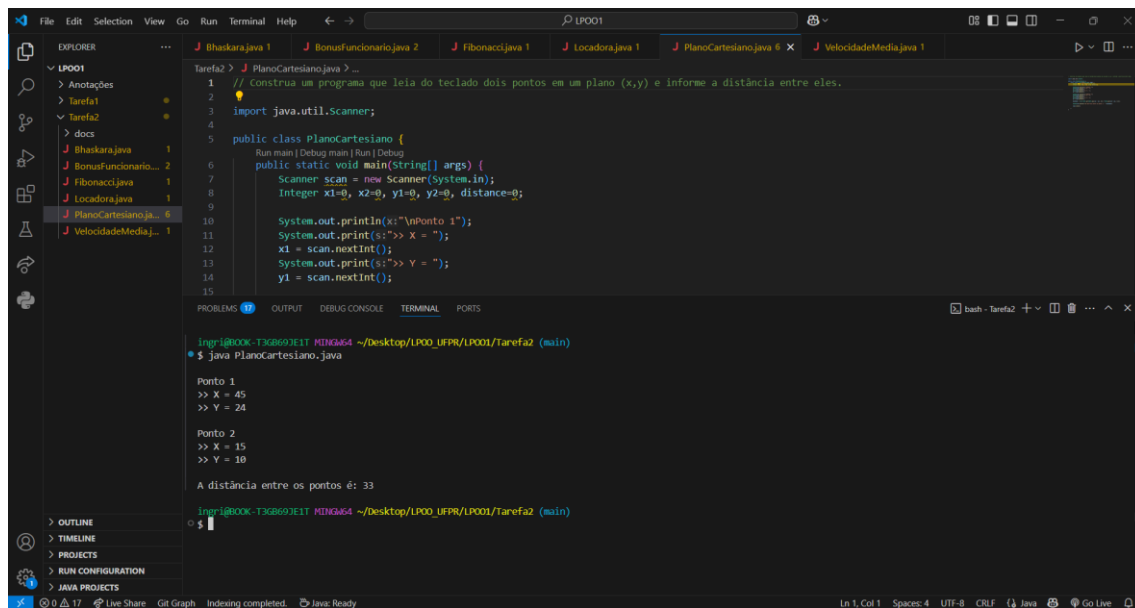
The screenshot shows the same IDE with the file explorer listing the Java files. The main editor displays the code for `Locadora.java`. The code imports `java.util.Scanner` and contains a `main` method that prompts the user for the number of DVDs and the rental value. It then calculates and prints the annual rental fee and the profit from fines. The terminal at the bottom shows the command `java Locadora.java` being executed, with the user inputting `50` for DVDs and `4,00` for the rental value. The program output shows the annual rental fee as `R$ 768,00` and the profit from fines as `R$ 0,40`.

```
14
15 import java.util.Scanner;
16
17 public class Locadora {
18     /**
19      * Método principal que executa os cálculos de faturamento e multas.
20      */
21     public static void main(String[] args) {
22         Scanner scan = new Scanner(System.in);
23         Integer qtd;
24         Float aluguel, faturamentoAnual, multasMensais, faturamentoMensal;
25
26         // Inputs necessários para fazer os cálculos
27         System.out.print("Informe a quantidade de DVDs: ");
28         qtd = scan.nextInt();
29     }
30 }
```

```
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java Locadora.java
Informe a quantidade de DVDs: 50
Informe o valor do aluguel: 4,00
Faturamento anual: R$ 768,00
Ganho com multas por mês: R$ 0,40
ingri@BOOK-T3G8693E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste

PlanoCartesiano.java



```
1 // Construa um programa que leia do teclado dois pontos em um plano (x,y) e informe a distância entre eles.
2
3 import java.util.Scanner;
4
5 public class PlanoCartesiano {
6     Run main | Debug main | Run | Debug
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9         Integer x1=0, x2=0, y1=0, y2=0, distance=0;
10
11         System.out.println(x1+"\nPonto 1");
12         System.out.print(s:">> X = ");
13         x1 = scan.nextInt();
14         System.out.print(s:">> Y = ");
15         y1 = scan.nextInt();
16
17         // ... (calculations for distance) ...
18     }
19 }
```

Terminal Output:

```
ingri@BOOK-T3G869J3E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java PlanoCartesiano.java

Ponto 1
>> X = 45
>> Y = 24

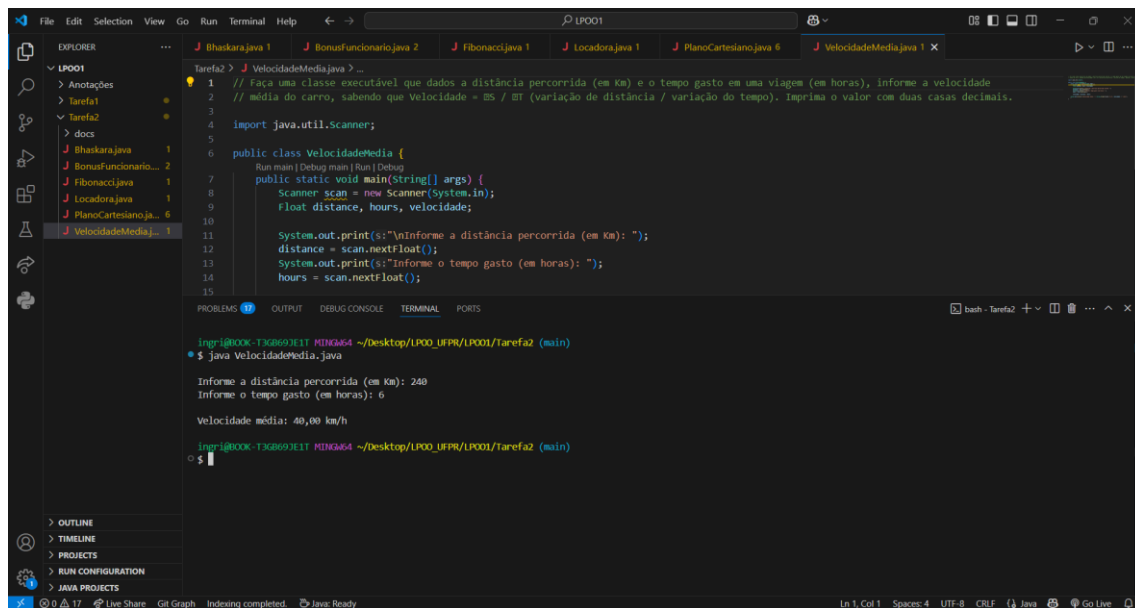
Ponto 2
>> X = 15
>> Y = 10

A distância entre os pontos é: 33

ingri@BOOK-T3G869J3E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste

VelocidadeMedia.java



```
1 // Faça uma classe executável que dados a distância percorrida (em Km) e o tempo gasto em uma viagem (em horas), informe a velocidade
2 // média do carro, sabendo que Velocidade = DS / DT (variação de distância / variação do tempo). Imprima o valor com duas casas decimais.
3
4 import java.util.Scanner;
5
6 public class VelocidadeMedia {
7     Run main | Debug main | Run | Debug
8     public static void main(String[] args) {
9         Scanner scan = new Scanner(System.in);
10         float distance, hours, velocidade;
11
12         System.out.print(s:"\nInforme a distância percorrida (em Km): ");
13         distance = scan.nextFloat();
14         System.out.print(s:"\nInforme o tempo gasto (em horas): ");
15         hours = scan.nextFloat();
16
17         // ... (calculations for average velocity) ...
18     }
19 }
```

Terminal Output:

```
ingri@BOOK-T3G869J3E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$ java VelocidadeMedia.java

Informe a distância percorrida (em Km): 240
Informe o tempo gasto (em horas): 6

Velocidade média: 40,00 km/h

ingri@BOOK-T3G869J3E1T MINGW64 ~/Desktop/LP00_UFPR/LP001/Tarefa2 (main)
$
```

Teste