



Lista de Exercícios III

Considerações Iniciais:

Esta lista de exercício deve:

- Ser realizada em equipes de até 06 alunos.
- Ser entregue no **prazo** proposto.
- Ter os algoritmos pedidos escritos em **linguagem Portugol, Java ou C#**.
- Ter todos os algoritmos **devidamente indentados**.

Exercícios:

1. Escreva um algoritmo que leia 3 números a partir do teclado.
O algoritmo deve apresentar:

- a. O maior número.
- b. O menor número.
- c. A média aritmética dos três números.

```
1 programa {
2   funcao inicio() {
3     inteiro num1, num2, num3, maior, menor, mediaaritmética
4     escreva("digite o primeiro numero: ") leia (num1)
5     escreva("\ndigite o segundo numero: ") leia (num2)
6     escreva("\ndigite o terceiro numero: ") leia (num3)
7
8     mediaaritmética = (num1 + num2 + num3) / 3
9     escreva("\na mediaaritmética foi de: " + mediaaritmética)
10
11    maior = num1
12    menor = num1
13
14    se (num2 > maior)
15      maior = num2
16    se (num2 < menor)
17      menor = num2
18
19    se (num3 > maior)
20      maior = num3
21    se (num3 < menor)
22      menor = num3
23
24    escreva("\no maior numero foi: " + maior)
25    escreva("\no menor numero foi: " + menor)
26  }
27}
```

digite o terceiro numero: 1

a mediaaritmética foi de: 2
o maior numero foi: 3
o menor numero foi: 1

Programa finalizado. Tempo de execução: 4246 milissegundos

2. Escreva o algoritmo de uma máquina de venda automática de salgadinhos, doces, sucos e refrigerantes. O algoritmo deve calcular o menor número de notas que deve ser dado de TROCO para um pagamento efetuado.

O algoritmo deve ler o valor da compra e o valor pago. Se o valor pago for menor que o valor da compra, a máquina deve apresentar uma mensagem, informando que a quantia paga é insuficiente para realizar a compra. A máquina aceita apenas notas de R\$ 50,00, R\$ 20,00, R\$ 10,00, R\$ 5,00, R\$ 2,00 e R\$ 1,00.

Exemplo:

Centro Universitário UNA

Programação de Soluções Computacionais

Professor Daniel Henrique Matos de Paiva



Valor pago: R\$ 100,00

Valor da compra: R\$ 23,00

Troco: R\$ 77,00

Notas de R\$ 50,00: 1

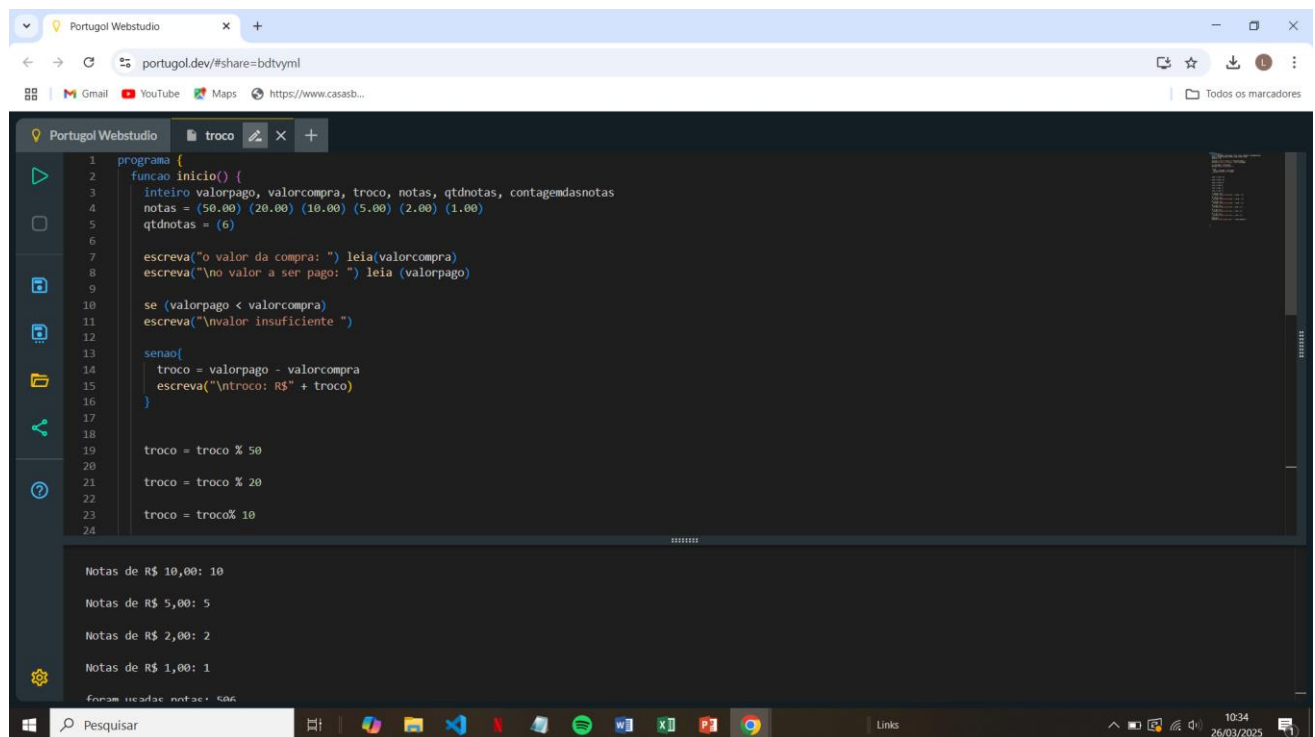
Notas de R\$ 20,00: 1

Notas de R\$ 10,00: 0

Notas de R\$ 5,00: 1

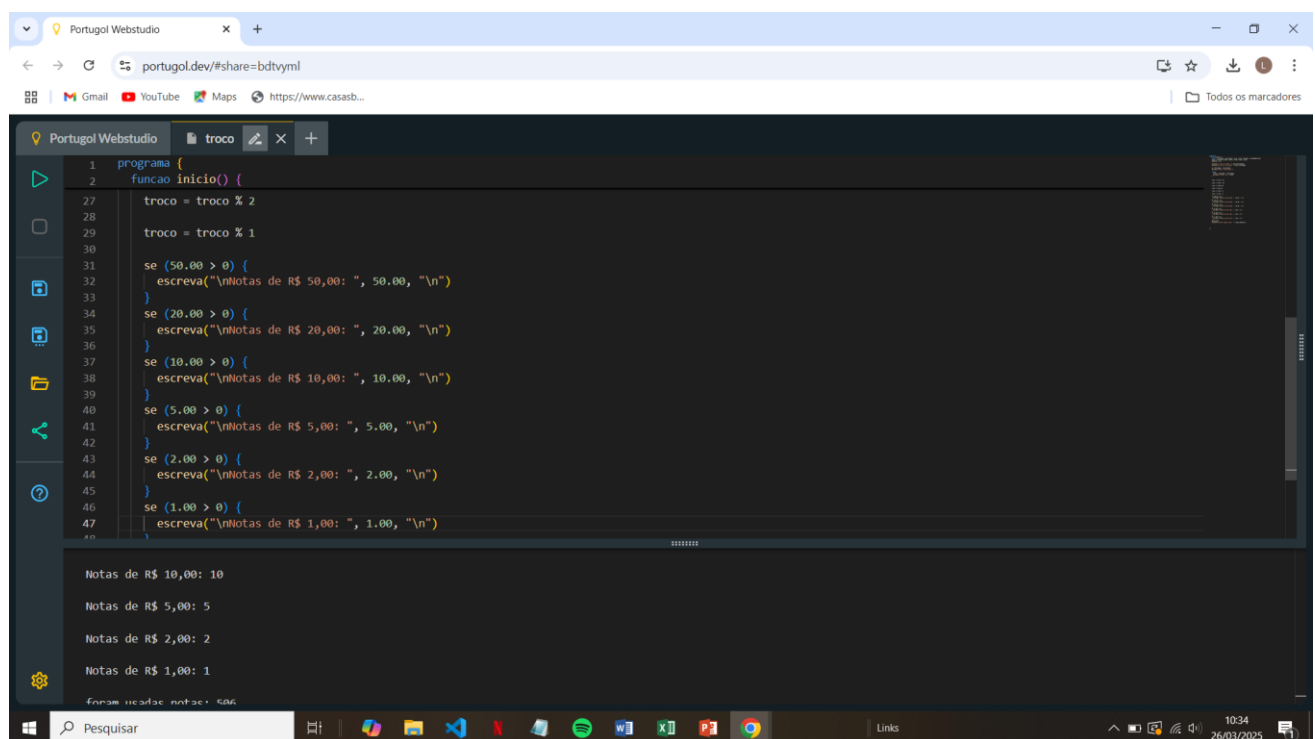
Notas de R\$ 2,00: 1

Notas de R\$ 1,00: 0



```
1 programa {
2   funcao inicio() {
3     inteiro valorpago, valorcompra, troco, notas, qtdnotas, contagemdasnotas
4     notas = (50.00) (20.00) (10.00) (5.00) (2.00) (1.00)
5     qtdnotas = (6)
6
7     escreva("o valor da compra: ") leia(valorcompra)
8     escreva("\no valor a ser pago: ") leia (valorpago)
9
10    se (valorpago < valorcompra)
11      escreva("\nvalor insuficiente ")
12
13    senao{
14      troco = valorpago - valorcompra
15      escreva("\ntroco: R$" + troco)
16    }
17
18    troco = troco % 50
19
20    troco = troco % 20
21
22    troco = troco % 10
23
24  }
```

Notas de R\$ 10,00: 10
Notas de R\$ 5,00: 5
Notas de R\$ 2,00: 2
Notas de R\$ 1,00: 1



```
1 programa {
2   funcao inicio() {
3     troco = troco % 2
4     troco = troco % 1
5
6     se (50.00 > 0) {
7       escreva("\nNotas de R$ 50,00: ", 50.00, "\n")
8     }
9     se (20.00 > 0) {
10      escreva("\nNotas de R$ 20,00: ", 20.00, "\n")
11    }
12    se (10.00 > 0) {
13      escreva("\nNotas de R$ 10,00: ", 10.00, "\n")
14    }
15    se (5.00 > 0) {
16      escreva("\nNotas de R$ 5,00: ", 5.00, "\n")
17    }
18    se (2.00 > 0) {
19      escreva("\nNotas de R$ 2,00: ", 2.00, "\n")
20    }
21    se (1.00 > 0) {
22      escreva("\nNotas de R$ 1,00: ", 1.00, "\n")
23    }
24  }
```

Notas de R\$ 10,00: 10
Notas de R\$ 5,00: 5
Notas de R\$ 2,00: 2
Notas de R\$ 1,00: 1

```
1 programa {
2   funcao inicio() {
34    se (20.00 > 0) {
36      }
37    se (10.00 > 0) {
38      escreva("\nNotas de R$ 10,00: ", 10.00, "\n")
39    }
40    se (5.00 > 0) {
41      escreva("\nNotas de R$ 5,00: ", 5.00, "\n")
42    }
43    se (2.00 > 0) {
44      escreva("\nNotas de R$ 2,00: ", 2.00, "\n")
45    }
46    se (1.00 > 0) {
47      escreva("\nNotas de R$ 1,00: ", 1.00, "\n")
48    }
49    qtdnotas>0
50    escreva("\nforam usadas notas: " + notas,qtdnotas )
51  }
52 }
53
54 }
55 }
56 }
```

Notas de R\$ 10,00: 10
Notas de R\$ 5,00: 5
Notas de R\$ 2,00: 2
Notas de R\$ 1,00: 1
foram usadas notas: 506

```
1 programa {
2   funcao inicio() {
34    se (20.00 > 0) {
36      }
37    se (10.00 > 0) {
38      escreva("\nNotas de R$ 10,00: ", 10.00, "\n")
39    }
40    se (5.00 > 0) {
41      escreva("\nNotas de R$ 5,00: ", 5.00, "\n")
42    }
43    se (2.00 > 0) {
44      escreva("\nNotas de R$ 2,00: ", 2.00, "\n")
45    }
46    se (1.00 > 0) {
47      escreva("\nNotas de R$ 1,00: ", 1.00, "\n")
48    }
49    qtdnotas>0
50    escreva("\nforam usadas notas: " + notas,qtdnotas )
51  }
52 }
53
54 }
55 }
56 }
```

o valor da compra: 30
o valor a ser pago: 100
troco: R\$70
Notas de R\$ 50,00: 50
Notas de R\$ 20,00: 20
Notas de R\$ 10,00: 10

3. Escreva um algoritmo que resolva uma equação de segundo grau, realizando a verificação de consistência dos valores dos coeficientes ("a", "b" e "c") e do discriminante (delta).

a. Se os coeficientes "a" e "b" forem iguais a zero e o coeficiente "c" for diferente de zero, apresentar a mensagem "Coeficientes informados incorretamente."

b. Caso o coeficiente "a" seja igual a zero e o coeficiente "b" for diferente de zero,

deverá ser impressa a mensagem: "Essa é uma equação de primeiro grau" e deverá ser informado o valor da raiz real da equação.

c. Caso o discriminante seja negativo, deverá ser impressa a mensagem: "Esta equação não possui raízes reais".

d. Caso o discriminante seja zero, apresentar a mensagem "Esta equação possui duas raízes reais iguais. " e informar o valor das raízes da equação.

e. Caso o discriminante seja maior que zero, apresentar a mensagem "Esta equação possui duas raízes reais diferentes. " e informar o valor das raízes da equação.

Centro Universitário UNA

Programação de Soluções Computacionais

Professor Daniel Henrique Matos de Paiva



Equação do segundo grau: $ax^2 + bx + c = 0$

Discriminante: $\Delta = b^2 - 4ac$

$$ax^2 + bx + c = 0$$
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Portugol Webstudio

portugol.dev

Sem título

```
1 programa {
2   funcao inicio() {
3     real valorA, valorB, valorC, valorX
4
5     escreva ("Digite o valor de a: ")
6     leia (valorA)
7     escreva ("Digite o valor de b: ")
8     leia (valorB)
9     escreva ("Digite o valor de c: ")
10    leia (valorC)
11
12    se (valorA == 0 e valorB == 0 e valorC != 0) {
13      escreva ("Coeficientes informados incorretamente. ")
14    }
15    senao se (valorA == 0 e valorB != 0) {
16      valorX = -valorC / valorB
17      escreva ("Essa é uma equação de primeiro grau. O valor da raiz real desta equação é: " + valorX)
18    }
19  }
20 }
21
22
```

Digite o valor de a: 0
Digite o valor de b: 0
Digite o valor de c: 1
Coeficientes informados incorretamente.
Programa finalizado. Tempo de execução: 8586 milissegundos

Pesquisar

Links 25°C Pred. limpo 2005 28/03/2025

Portugol Webstudio

Novo separador

portugol.dev

Sem título

```
1 programa {
2   real valorA, valorB, valorC, valorX
3
4   escreva ("Digite o valor de a: ")
5   leia (valorA)
6   escreva ("Digite o valor de b: ")
7   leia (valorB)
8   escreva ("Digite o valor de c: ")
9   leia (valorC)
10
11   se (valorA == 0 e valorB == 0 e valorC != 0) {
12     escreva ("Coeficientes informados incorretamente. ")
13   }
14
15   senao (valorA == 0 e valorB != 0)
16     escreva ("Essa é uma equação de primeiro grau. ")
17
18   valorX = -valorC / valorB
19   se (valorX < 0) {
20     escreva ("Esta equação não possui raízes reais. ")
21   }
22   senao {
23     escreva ("O valor da raiz real desta equação é: " + valorX)
24   }
25
26 }
27
28
```

Digite o valor de a: 0
Digite o valor de b: 2
Digite o valor de c: 8
Essa é uma equação de primeiro grau. Esta equação não possui raízes reais.
Programa finalizado. Tempo de execução: 4842 milissegundos

Pesquisar

Links 25°C Pred. limpo 2009 28/03/2025

```
17 }
18
19 valorX = -valorC / valorB
20 escreva ("A raiz da equação é: " + valorX)
21
22 se (valorX < 0) {
23   escreva (" Esta equação não possui raízes reais. ")
24 }
25 senao {
26   escreva ("Essa é uma equação de segundo grau.")
27   delta = (valorB * valorB) - (4 * valorA * valorC)
28 }
29 se (delta < 0) {
30   escreva ("Essa equação não possui raízes reais. ")
31 }
32 senao se (delta == 0) {
33   valorX1 = -valorB / (2 * valorA)
34   escreva ("Esta equação possui duas raízes reais iguais: "+ valorX1)
35 }
36 senao {
37   valorX1 = (-valorB + raiz_quadrada(delta)) / (2 * valorA)
38   valorX2 = (-valorB - raiz_quadrada(delta)) / (2 * valorA)
39   escreva("As raízes da equação são: x1 = ", valorX1, " e x2 = ", valorX2, "\n")
40 }
41 }
42 }
43
```

Digite o valor de b: 6
Digite o valor de c: 6
A raiz da equação é: -1
Esta equação não possui raízes reais.
Função não declarada: raiz_quadrada

Programa finalizado. Tempo de execução: 16922 milissegundos

4. Escreva um algoritmo que leia dois valores: o primeiro servindo de indicador de operação e o segundo correspondendo ao raio de um círculo ou esfera. Caso o primeiro valor lido seja:

- 1: calcular e imprimir o perímetro do círculo.
- 2: calcular e imprimir a área do círculo.
- 3: calcular e imprimir o volume da esfera.

Se o primeiro valor lido for diferente desses três valores possíveis, imprimir uma mensagem de erro, informando que o código da operação é inválido.

OBS:

Perímetro de um círculo: $2 \pi r$

Área de um círculo: πr^2

Volume de uma esfera: $\frac{4}{3} \pi r^3$

em que $\pi = 3,141592$ e r é o raio da circunferência.

```
1 programa {
2   funcao inicio() {
3     real PI = 3.14
4     inteiro opcao, raio, resultado
5
6     escreva("digite o codigo da operação (1-perimetro 2-area 3-volume): ") leia(opcao)
7     escreva("\n o raio: ") leia(raio)
8
9     se (opcao == 1) {
10      resultado = 2 * PI * raio
11      escreva("\no perimetro do circulo é: " + resultado)
12    }
13    se (opcao == 2) {
14      resultado = PI * (raio*raio)
15      escreva("\na area do circulo é: " + resultado)
16    }
17    se (opcao == 3){
18      resultado = 4/3 * PI * (raio*raio)
19      escreva("\no volume do circulo é: " + resultado)
20    }
21  }
22 }
23
24
```

digite o codigo da operação (1-perimetro 2-area 3-volume): 1

o raio: 1

o perimetro do circulo é: 6.28

Programa finalizado. Tempo de execução: 3561 milissegundos

5. Escrever um algoritmo que leia a partir do teclado dois números reais e um dos símbolos de operação: + , - , * , / ou ^ .

O algoritmo deve retornar o resultado da operação.

Caso o símbolo informado seja diferente dos símbolos preestabelecidos, o algoritmo deve apresentar uma mensagem de erro, informando que o símbolo da operação é inválido.

```
1 programa {
2   funcao inicio() {
3     real num1, num2
4     caracter operacao
5     real resultado
6
7     escreva("digite o primeiro numero: ")
8     leia(num1)
9     escreva("\ndigite o segundo numero: ")
10    leia(num2)
11    escreva("\ndigite a operacao: ")
12    leia(operacao)
13
14    se (operacao != "+" e operacao != "-" e operacao != "/" e operacao != "^")
15      escreva("operador invalido")
16    senao{
17      escolha (operacao){
18        caso "+":
19          resultado = num1 + num2
20      }
21    }
22    escreva("o resultado sera: " + resultado)
23  }
24 }
25
26
```

digite o segundo numero: 2

digite a operacao: *

o resultado sera: 4

Programa finalizado. Tempo de execução: 3566 milissegundos

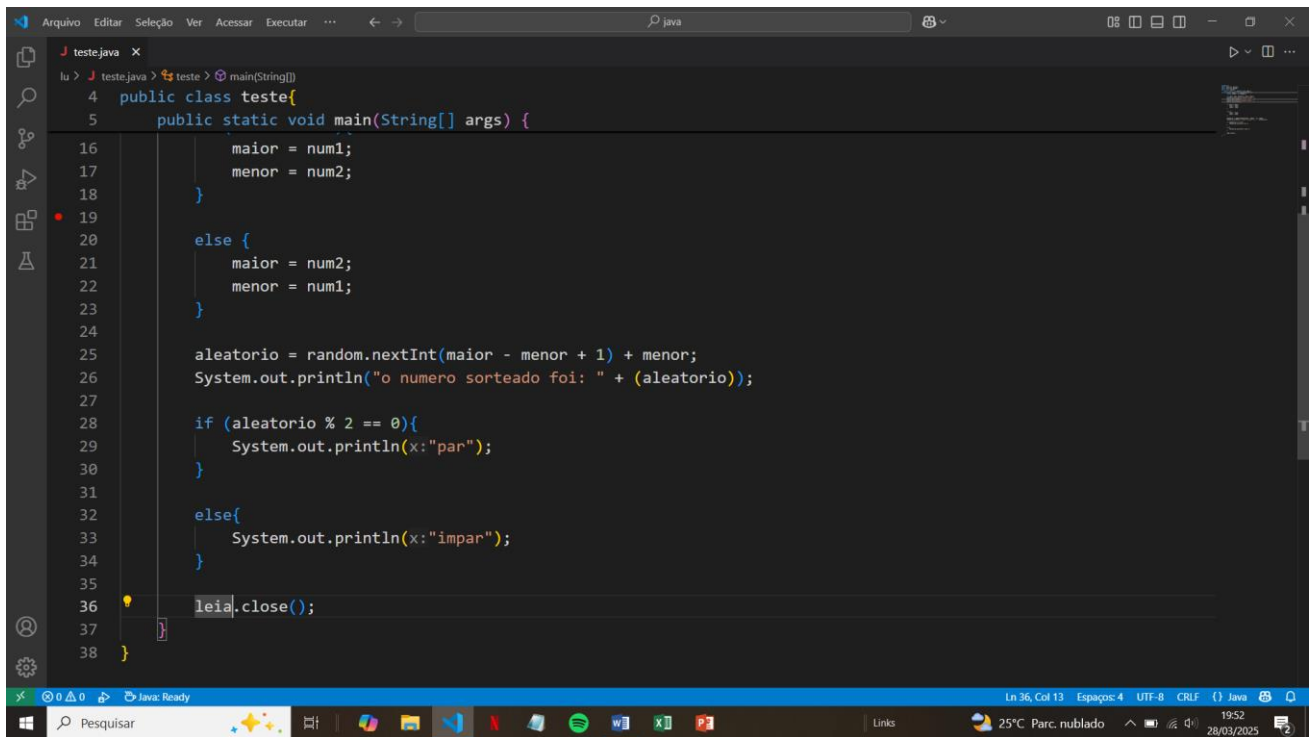


6. Crie um algoritmo que leia dois números inteiros e realize o sorteio de um número aleatório. O algoritmo deve validar qual é o **menor** e o **maior** número informado pelo teclado, para que independente da ordem que o usuário digite os números, consiga realizar o sorteio.

Se o algoritmo gerar um número par, escreva na tela o número gerado e informe que ele é par. Se o algoritmo gerar um número ímpar, escreva na tela o número gerado e que ele é um número ímpar.

A screenshot of an IDE window showing a Java file named 'teste.java'. The code imports Scanner and Random, defines a main method, reads two integers, compares them to find the min and max, and then generates a random number within that range. The IDE interface includes a menu bar, a toolbar, a sidebar with project and file views, and a Windows taskbar at the bottom showing various application icons and system status information like temperature and date.

```
1  teste.java
2  import java.util.Scanner;
3  import java.util.Random;
4  public class teste{
5      Run | Debug
6      public static void main(String[] args) {
7          Scanner leia = new Scanner(System.in);
8          Random random = new Random();
9
10         int num1, num2, aleatorio, maior, menor;
11         System.out.println(x:"primeiro numero: ");
12         num1 = leia.nextInt();
13         System.out.println(x:"segundo numero: ");
14         num2 = leia.nextInt();
15
16         if (num1 > num2){
17             maior = num1;
18             menor = num2;
19         }
20
21         else {
22             maior = num2;
23             menor = num1;
24         }
25
26         aleatorio = random.nextInt(maior - menor + 1) + menor;
27         System.out.println("o numero sorteado foi: " + (aleatorio));
28     }
29 }
```



```
1  teste.java x
2  lu > J teste.java > teste > main(String[])
3
4  public class teste{
5      public static void main(String[] args) {
6
7          // ...
8
9          maior = num1;
10         menor = num2;
11     }
12
13     else {
14         maior = num2;
15         menor = num1;
16     }
17
18     aleatorio = random.nextInt(maior - menor + 1) + menor;
19     System.out.println("o numero sorteado foi: " + (aleatorio));
20
21     if (aleatorio % 2 == 0){
22         System.out.println(x:"par");
23     }
24
25     else{
26         System.out.println(x:"impar");
27     }
28
29     leia.close();
30 }
31
32
33
34
35
36
37
38 }
```

7. Suba os algoritmos realizados para o GitHub com o nome do repositório: “psc lista-03-2025”.