

Lista 1 – LPOO_I

1. Determine as raízes de uma equação de 2º grau: $ax^2 + bx + c = 0$ (recordar que o discriminante $\Delta = b^2 - 4ac$, e que a raiz $r = (-b \pm \sqrt{\Delta})/2a$).
2. Calcule a distância entre dois pontos num espaço de 3 dimensões.
3. O programa a seguir estranhamente sempre escreve “A distancia e: 1.0”. Identifique onde está o defeito.

```
import java.util.Scanner;

public class Distancia {
    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        double x1, y1, x2, y2, distancia;

        System.out.println("Entre com as coordenadas x e y dos pontos nesta or
        x1 = teclado.nextFloat();
        y1 = teclado.nextFloat();
        x2 = teclado.nextFloat();
        y2 = teclado.nextFloat();

        distancia = Math.pow(Math.pow(x2-x1, 2) + Math.pow(y2-y1, 2), 1/2);
        System.out.println("A distância é: " + distancia);
    }
}
```

4. Para cada produto informado (nome, preço e quantidade), escreva o nome do produto comprado e o valor total a ser pago, considerando que são oferecidos descontos pelo número de unidades compradas, segundo a tabela abaixo:
 - a. Até 10 unidades: valor total
 - b. de 11 a 20 unidades: 10% de desconto
 - c. de 21 a 50 unidades: 20% de desconto
 - d. acima de 50 unidades: 25% de desconto
5. Construa a tabela de multiplicação de números de 1 a 10 (ex.: $1 \times 1 = 1$, $1 \times 2 = 2$, etc.).
6. Determine o número de dígitos de um número informado.
7. Considere os programas a seguir, que leem um código repetidamente e imprimem o código lido até que o código lido seja igual a -1. O código -1 não deve ser impresso.
 - a. Qual das duas é a correta
 - b. Como corrigir a incorreta?

Programa A

```
import java.util.Scanner;

public classCodigo {
    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int codigo;

        System.out.println("Informe o código: ");
        codigo = teclado.nextInt();
        while (codigo != -1) {
            System.out.println("Código: " + codigo);
            System.out.println("Informe o código: ");
            codigo = teclado.nextInt();
        }
    }
}
```

Programa B

```
import java.util.Scanner;

public classCodigo {
    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int codigo;

        do {
            System.out.print("Informe o código: ");
            codigo = teclado.nextInt();
            System.out.println("Código: " + codigo);
        } while (codigo != -1);
    }
}
```

8. Calcule a série de Fibonacci para um número inteiro não negativo informado pelo usuário. A série de Fibonacci inicia com os números $F_0 = 0$ e $F_1 = 1$, e cada número posterior equivale à soma dos dois números anteriores ($F_n = F_{n-1} + F_{n-2}$). Por exemplo, caso o usuário informe o número 9, o resultado seria: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34.
9. Determine a saída do seguinte programa:

```
public class Loop {
    public static void main(String[] args) {
        for (int i = 2; i <= 8; i = i + 2) {
            for (int j = i; j <= 4; j++) {
                for (int k = 1; k <= j; k = k + 1) {
                    System.out.println(i + ", " + j + ", " + k);
                }
            }
        }
    }
}
```

10. Calcule a raiz quadrada aproximada de um número inteiro informado pelo usuário, respeitando o erro máximo também informado pelo usuário. Não utilize funções predefinidas.
11. Leia uma matriz 3 x 3 que representa um tabuleiro de jogo da velha e indique qual posição deveria ser jogada para ganhar o jogo (se possível) ou ao menos para evitar uma derrota.
12. Ordene um vetor de 100 números inteiros gerados aleatoriamente.
13. Crie dois vetores de 50 posições com valores inteiros aleatórios, ordene cada vetor individualmente, e combine os dois vetores gerando um novo vetor de 100 posições, de forma que esse novo vetor já seja criado ordenado.