

26. [Sebesta, 2000] Escreva um tipo de dado abstrato para números complexos, incluindo operações de adição, de subtração, de multiplicação, de divisão, de extração de cada uma das partes de um número complexo e de construção de um número complexo a partir de duas constantes, variáveis ou expressões de ponto flutuante. Use o Modula-2, a Ada, o C++ ou o Java.

```
/**
 * Classe Java responsavel pela representacao de um numero complexo
 */
public class Complex
{
    /**
     * Parte real do numero complexo
     */
    private final double realPart;

    /**
     * Parte imaginaria do numero complexo
     */
    private final double imaginaryPart;

    /**
     * Construtor para inicializar o numero complexo
     */
    public Complex()
    {
        this(0.0, 0.0);
    }

    /**
     * Construtor para inicializar a parte real e a parte imaginaria
     * do numero complexo
     *
     * @param realPart parte real do numero complexo
     * @param imaginaryPart parte imaginaria do numero complexo
     */
    public Complex(final double realPart, final double imaginaryPart)
    {
        this.realPart = realPart;

        this.imaginaryPart = imaginaryPart;
    }

    /**
     * Retornar a parte real do numero complexo
     *
     * @return parte real do numero complexo
     */
    public double getRealPart()
    {
        return realPart;
    }
}
```

```
/**
 * Retornar a parte imaginaria do numero complexo
 *
 * @return parte imaginaria do numero complexo
 */
public double getImaginaryPart()
{
    return imaginaryPart;
}

/**
 * Retornar um novo objeto Complex cujo valor eh this + other
 *
 * @param other numero complexo a ser adicionado
 * @return novo objeto Complex cujo valor eh this + other
 */
public Complex addition(final Complex other)
{
    final var real = realPart + other.realPart;

    final var imaginary = imaginaryPart + other.imaginaryPart;

    return new Complex(real, imaginary);
}

/**
 * Retornar um novo objeto Complex cujo valor eh this - other
 *
 * @param other numero complexo a ser subtraido
 * @return novo objeto Complex cujo valor eh this - other
 */
public Complex subtraction(final Complex other)
{
    final var real = realPart - other.realPart;

    final var imaginary = imaginaryPart - other.imaginaryPart;

    return new Complex(real, imaginary);
}

/**
 * Retornar um novo objeto Complex cujo valor eh this * other
 *
 * @param other numero complexo a ser multiplicado
 * @return novo objeto Complex cujo valor eh this * other
 */
public Complex multiplication(final Complex other)
{
    final var real = (realPart * other.realPart) +
                    (imaginaryPart * other.imaginaryPart * (-1));

    final var imaginary = (realPart * other.imaginaryPart) +
                        (imaginaryPart * other.realPart);

    return new Complex(real, imaginary);
}
```

```
/**
 * Retornar um novo objeto Complex cujo valor eh this / other
 *
 * @param other numero complexo a ser dividido
 * @return novo objeto Complex cujo valor eh this / other
 */
public Complex division(final Complex other)
{
    final var real = ((realPart * other.realPart) +
                     (imaginaryPart * other.imaginaryPart)) /
                    ((other.realPart * other.realPart) +
                     (other.imaginaryPart * other.imaginaryPart));

    final var imaginary = ((other.realPart * imaginaryPart) -
                          (realPart * other.imaginaryPart)) /
                          ((other.realPart * other.realPart) +
                           (other.imaginaryPart * other.imaginaryPart));

    return new Complex(real, imaginary);
}

@Override
public String toString()
{
    return "(" + realPart + ", " + imaginaryPart + ")";
}

/**
 * Metodo principal da linguagem de programacao Java
 *
 * @param args argumentos da linha de comando (nao utilizado)
 */
public static void main(final String[] args)
{
    final var scanner = new Scanner(System.in);

    System.out.println("Forneça o primeiro número complexo");

    System.out.print("Valor da parte real: ");

    var realPart = scanner.nextDouble();

    System.out.print("Valor da parte imaginária: ");

    var imaginaryPart = scanner.nextDouble();

    final var complex1 = new Complex(realPart, imaginaryPart);

    System.out.println("Forneça o segundo número complexo");

    System.out.print("Valor da parte real: ");

    realPart = scanner.nextDouble();

    System.out.print("Valor da parte imaginária: ");

    imaginaryPart = scanner.nextDouble();

    final var complex2 = new Complex(realPart, imaginaryPart);
```

```
scanner.close();

final var complex3 = complex1.addition(complex2);
final var complex4 = complex1.subtraction(complex2);
final var complex5 = complex1.multiplication(complex2);
final var complex6 = complex1.division(complex2);

System.out.println(complex1 + " + " + complex2 + " = " + complex3);
System.out.println(complex1 + " - " + complex2 + " = " + complex4);
System.out.println(complex1 + " * " + complex2 + " = " + complex5);
System.out.println(complex1 + " / " + complex2 + " = " + complex6);
}
```