

**Curso Técnico em Desenvolvimento de Sistemas**

**Disciplina:** Orientação a Objetos

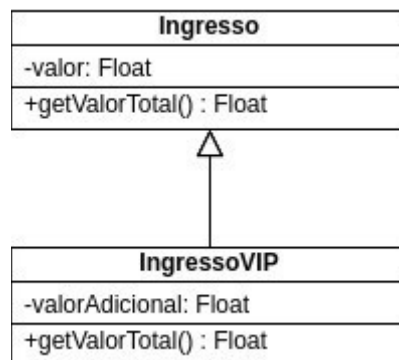
**Carga Horária:** 120 horas aula

**Período:** 2º ano

Data: 16/10/2023

**Exercícios – Orientação a objetos e encapsulamento**

1. Crie um programa em Java que implemente as classes **Ingresso** e **IngressoVIP** conforme o modelo abaixo. Perceba que existe uma relação de herança entre as classes, bem como que o método `getValorTotal()` é sobrescrito na classe filha.



Após, faça:

- a) Crie um objeto para cada tipo de classe, imprimindo os valores de cada tipo de ingresso.
- b) Crie uma nova classe denominada **IngressoCamarote** (filha de **IngressoVIP**), sendo que a mesma deve ter um atributo `valorAdicionalCamarote` e um método `getValorTotal()`. O valor do **IngressoCamarote** deve considerar o valor do **IngressoVIP** mais o seu valor adicional.

2. Um usuário decidiu fazer um programa orientado a objetos para resolver operações matemáticas utilizando polimorfismo. As operações que serão contempladas pelo programa são soma, subtração, multiplicação, divisão e resto. Neste sentido, faça um programa que:

- a) Declare uma classe Calculadora que possui como atributos dois números (*numA* e *numB*) e um método *calcular()*.
- b) Declare as classes Soma, Subtracao, Multiplicacao, Divisao e Resto que herdam de Calculadora e sobrescrevem o método *calcular()*, retornando como resultado o valor da operação matemática correspondente à classe.
- c) Execute uma rotina para ler os números (*numA* e *numB*) e a operação, exibindo o resultado do cálculo utilizando a classe pertinente. O programa deve parar de solicitar números quando for informado 0 e 0 para os números *numA* e *numB*.

**3. DESAFIO:** A criptografia foi um recurso criado pela humanidade com o intuito de trocar mensagens de forma a evitar que pessoas não autorizadas (como inimigos) pudessem compreendê-las. Neste sentido, o texto de uma mensagem pode ser criptografado utilizando um algoritmo, tornando sua mensagem ilegível. Somente após o texto ser descriptografado que a mensagem torna-se legível novamente.

Um algoritmo muito comum para realizar a criptografia de mensagens é a Cifra de César. Ele consiste em deslocar os caracteres de uma mensagem utilizando uma chave numérica. Abaixo, está detalhado um exemplo do uso da Cifra de César com a chave 3:

Normal: ABCDEFGHIJKLMNOPQRSTUVWXYZ  
Cifrado: DEFGHIJKLMNOPQRSTUVWXYZABC

É possível implementar um programa capaz de criptografar/descriptografar textos por meio da Cifra de Cesar utilizando Orientação a Objetos. Neste sentido, pode-se utilizar a seguinte estrutura:

- Classe abstrata CifraCesar: atributo do tipo String contendo as letras do alfabeto, bem como os métodos abstratos *criptografar()* e *descriptografar()*.

- Classes filhas, sendo: CifraCesar4, CifraCesar8 e CifraCesarM5, onde devem ser sobrescritos os métodos *criptografar()* e *descriptografar()* de acordo com a chave (4, 8 e -5, respectivamente).

Por fim, implemente uma classe de execução que apresente um menu para o usuário, onde ele deve escolher:

(1) Criptografar: ler um texto, ler o algoritmo de criptografia (CifraCesar4, CifraCesar8 e CifraCesarM5) e exibir a mensagem criptografada.

(2) Descriptografar: ler um texto criptografado, ler o algoritmo de criptografia (CifraCesar4, CifraCesar8 e CifraCesarM5) e exibir a mensagem descriptografada.

(0) Sair: encerrar o programa.

#### Dicas:

1. Pode-se percorrer os caracteres de uma String utilizando um laço:

1.1- Método `length()` retorna o tamanho da String.

1.2- Método `charAt(indice)` retorna o caractere existente no índice do parâmetro.

2. Parece obter o índice de um caractere dentro de um String, utilize o método `indexOf(caractere)`. Tal método retornará o valor -1 caso o caractere não seja encontrado na String.

3. É necessário implementar uma lógica para tratar o início e final do alfabeto.

4. Nesta versão inicial, utilize apenas palavras sem acentos.

5. Utilize como testes as palavras cifradas:

Chave 4: EPYRSW HS XHW, FIQ ZMRHSW E HMWGMTPMRE HI SVMIRXEGES E SFNIXSW!

Chave 8: ITCVWA LW BLA, JMU DQVLWA I LQAKQXTQVI LM WZQMVBKIWI I WJRMBA!

Chave -5: VGPIJN YJ OYN, WZH QDIYJN V YDNXDKGDIV YZ JMDZIOVXVJ V JWEZOJN!