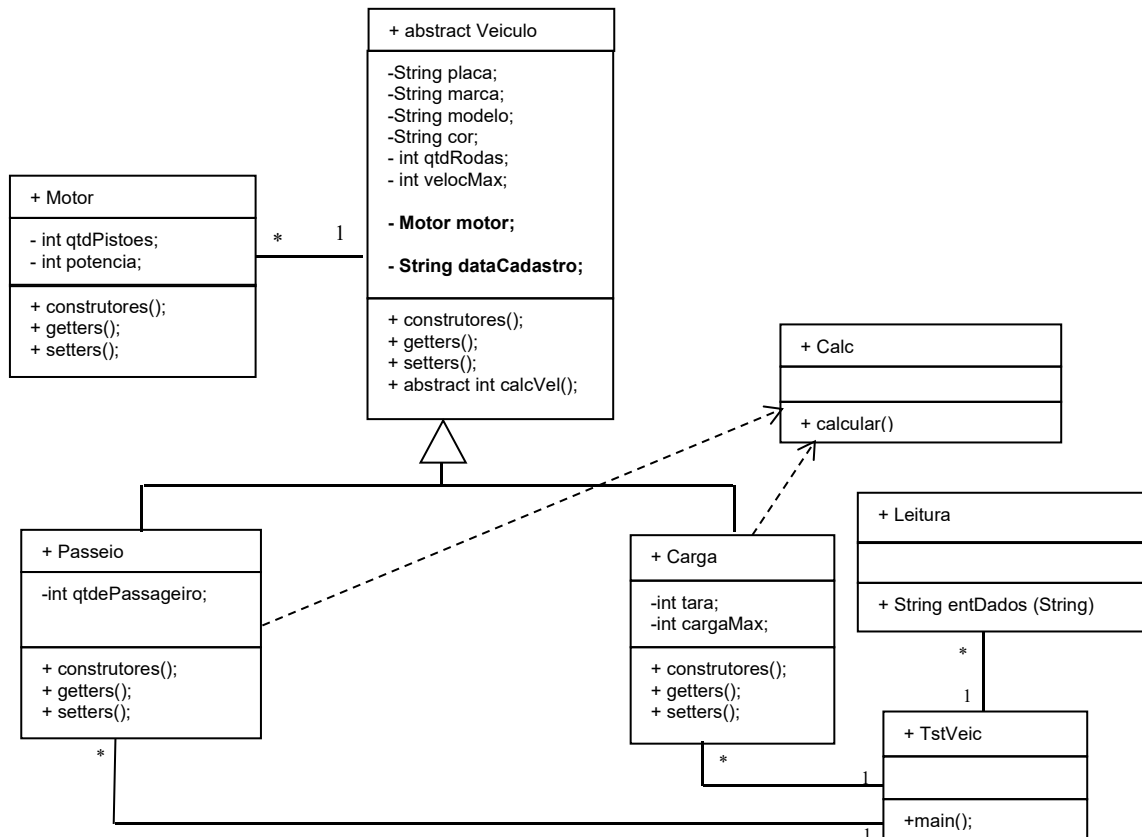


4ª Lista de Exercícios

Forma de entrega:

- TODAS as classes devem conter, nas primeiras linhas, o nome;
- **NÃO ENVIEM OS BYTECODES (.class)** - Compacte a pasta do projeto que contém os códigos fontes (arquivos.java) e nomeie o arquivo compactado como: *POO_ano-semester_nome do aluno*
Exemplo: **POO_2021-2_Cebolinha.rar** (ou .zip))



SUBSÍDIOS

- A “entrada” e o “armazenamento” da velocidade máxima dos veículos **sempre serão em Km/h**;
- Caso a velocidade máxima de um veículo seja menor que 10 e maior 250, deverá informar ao usuário que: “A velocidade está fora dos limites brasileiros”. Neste caso, após informar o usuário, deverá atribuir 100 Km/h, tanto para veículo de passeio quanto para veículo de carga.

Embora as velocidades **sempre sejam armazenadas em Km/h**, porém, **apenas para efeito de exibição**, por meio da classe **TstVeic**:

- a velocidade do carro de **passeio** deverá ser calculada e exibida em M/h (metros por hora);
- a velocidade do **caminhão** (veículo de carga) deverá ser calculada e exibida em Cm/h (centímetros por hora);

Obs.: Para isso o método abstrato **calcVel**, da classe **Veiculo**, **deverá retornar** o resultado do cálculo e este exibido na classe **TstVeic**;

- Perceba que, pelo diagrama não há uma associação entre as classes **Veiculo** e **TstVeic** nem entre **Motor** e **TstVeic**, todavia deverá instanciar motor (dos Veículos), na classe **TstVeic**, **sem declarar um objeto do tipo motor nas classes**: **Passeio**, **Carga** e **TstVeic**. Somente haverá declaração de **Motor** na classe **Veiculo**;
- O método “**calcular**” da **interface “Calc”** deverá retornar um valor que será exibido pela classe **TstVeic**. Este valor, para cada objeto impresso, será:
 - Na classe **Passeio**: será soma das quantidades de letras existentes em todos os atributos do tipo **String**;
 - Na classe **Carga**: será a soma de todos os valores de todos contidos em todos os atributos numéricos;

Obs.: No Moodle, veja a “Aula5” do tópico “Materiais para Aula”. Contém os slides e vídeo sobre como se implementa a estrutura de dados denominada “interface”. Trata-se da aplicação do conceito de Polimorfismo por Sobrescrita e este conceito já foi visto em sala.

- **ENTRADA DE DADOS (deverá utilizar a classe Leitura):**
 - a classe **Leitura** será utilizada como interface de interação entre o usuário humano e o sistema. Conterá um método chamado **entDados**. Este método:
 - receberá uma **String** como parâmetro: esta será impressa na tela como informação ao usuário de que dado do veículo ele está informando ao sistema;
 - retornará uma **String**: valor informado pelo usuário para aquele dado.
- **TODAS as impressões deverão ser feitas na classe TstVeic.**
- **NÃO PERMITA QUE:**
 - QUE SEJA POSSÍVEL INSTANCIAR OBJETOS DIRETOS DA CLASSE MÃE (**Veiculo**)
 - SOBRESCREVAM OS MÉTODOS SETTERS DE **TODAS AS CLASSES**;
 - SOBRESCREVAM **NENHUM DOS MÉTODOS DAS CLASSES FILHAS** (**Passeio** e **Carga**);
 - QUE AS CLASSES FILHAS (**Passeio** e **Carga**) SEJAM HERDADAS.

E ainda que:

- OS MÉTODOS CONSTRUTORES DEVERÃO INICIALIZAR OS ATRIBUTOS DE SUA CLASSE DE ACORDO COM SEUS TIPOS DE DADOS.