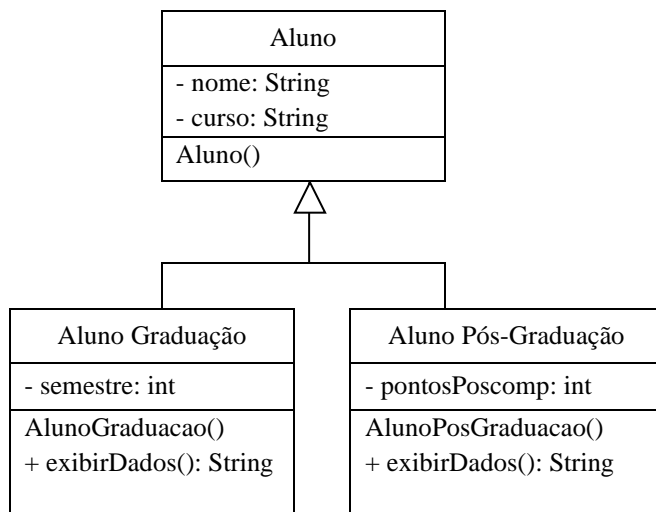




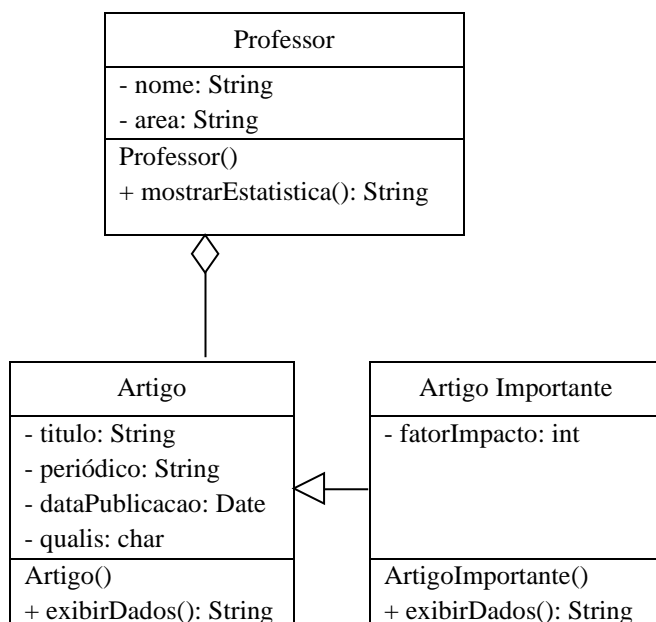
LISTA DE EXERCÍCIO – HERANÇA e POLIMORFISMO

1. O departamento de Sistemas e Computação da FURB está desenvolvendo um sistema para controlar os dados de seus alunos, tanto da graduação quanto da pós-graduação. Para tanto, definiu a seguinte estrutura de classes:



- A classe **Aluno** é classe-base para as outras duas. Portanto, **Aluno Graduação** e **Aluno Pós-Graduação** herdam as características de **Aluno**;
- O método `exibirDados` é implementado apenas nas classes descendentes. Cada uma responde a este método exibindo todos os dados que possui (herdados ou próprios);
- A interface ao usuário solicitará os dados dos alunos, questionando se é de **Graduação** ou **Pós-Graduação**, para então criar o objeto correspondente;
- Ao final, exibir os dados completos de todos os alunos digitados.

2. O departamento de Sistemas e Computação da FURB pretende levantar dados estatísticos sobre as publicações realizadas por seus professores. Um professor publica **Artigo**. Se o artigo é aceito em um editorial respeitado (Ex.: *Siggraph*, *Nature*, *Science Magazine*, etc...), ele é considerado um **Artigo Importante**. Para tanto está sendo desenvolvido um sistema com a seguinte estrutura de classes:



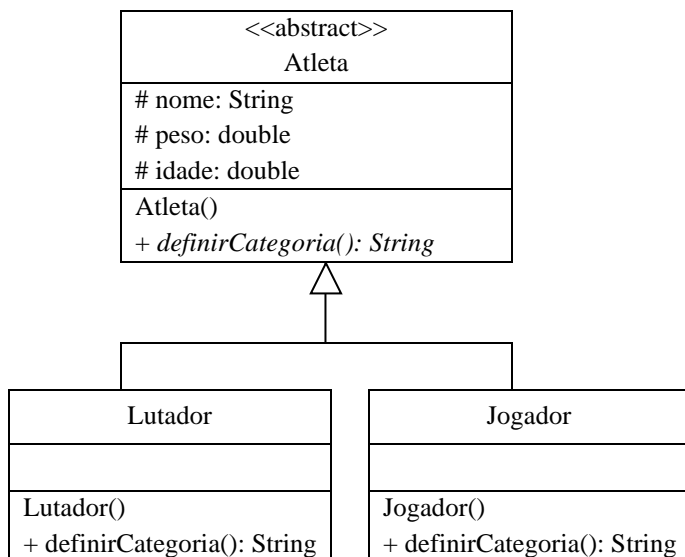
- A classe **Artigo** é classe-base para a classe **Artigo Importante**;
- A classe **Artigo** é parte da classe **Professor**;
- O método `exibirDados` é implementado apenas nas classes descendentes. Cada uma responde a este método exibindo todos os dados que possui (herdados ou próprios);
- A interface com o usuário questionará se o artigo será cadastrado como **Normal** ou **Importante**, para então criar o objeto correspondente;
- Ao final, mostrar os dados estatísticos de todos os professores, com os dados dos respectivos artigos publicados.

3. Numa academia, os atletas são categorizados pelo peso, conforme a tabela Atleta. Os lutadores, porém, são categorizados de maneira diferente, seguindo a tabela Lutador.

| Jogador | |
|---------------------|-----------|
| Faixa de idade | Categoria |
| até 16 | Infantil |
| acima de 16, até 18 | Juvenil |
| acima de 18 | Adulto |

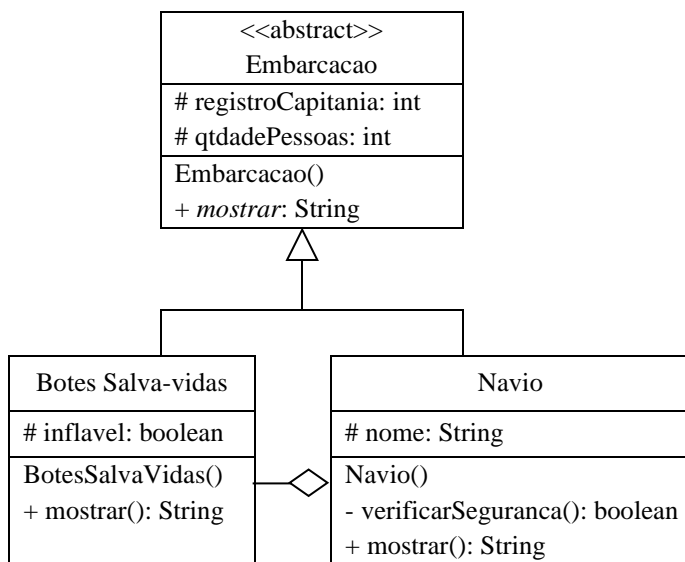
| Lutador | |
|---------------------|-----------|
| Faixa de Peso | Categoria |
| até 54kg | Pluma |
| acima de 54, até 60 | Leve |
| acima de 60, até 75 | Meio-leve |
| acima de 75 | Pesado |

Temos então, a seguinte hierarquia de classes:



- A classe Atleta é classe-base para a classe Lutador e Jogador. Portanto, Lutador e Jogador herdam as características de Atleta;
- A interface com o usuário solicitará os dados dos atletas, questionando se é Lutador ou Jogador, para então criar o objeto correspondente;
- Obs.: Armazene no mínimo um objeto de cada tipo (Lutador/Jogador);
- Ao final, exibir os dados completos de todos os atletas digitados e armazenados. Inclusive a categoria ao qual ele pertence, conforme faixa de peso e idade.

4. A agência de viagens XYZ oferece cruzeiros para diferentes destinos. Porém, com os recentes problemas de segurança, decidiu controlar/fiscalizar as condições dos navios utilizados nas viagens em relação ao equipamento básico de segurança: bote salva-vidas. Cada navio deve possuir botes salva-vidas em quantidade suficiente, capazes de acomodar todas as pessoas do navio. Para facilitar o controle, a agência decidiu contratar uma empresa para desenvolver um sistema com a seguinte estrutura de classes:



- A classe Embarcacao é classe-base para as classes Bote Salva-vidas e Navio. Portanto, estas classes herdam as características de Embarcacao;
- A classe Bote Salva-vidas é parte da classe Navio (agregação), pois um navio contém um ou vários botes.

Depois de armazenar vários navios e seus respectivos botes, deve ser possível digitar o número de registro de um Navio, imprimindo na tela seus dados juntamente com a mensagem “Navio atende normas de segurança” caso a soma das capacidades dos botes seja igual ou superior à capacidade de pessoas do navio. Caso contrário,

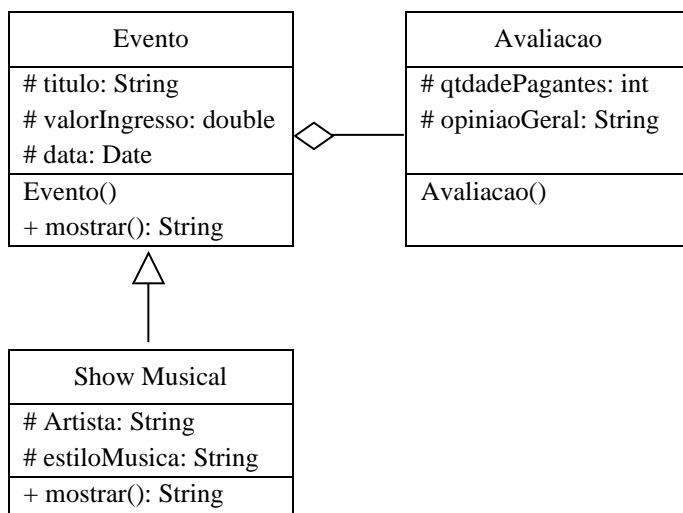
deve-se imprimir a mensagem “Navio NÃO atende normas de segurança, número de botes insuficiente”.

Exemplo de dados:

| | | | |
|--|--|--|---|
| Reg. Capitania: 768 Qtde Pessoas: 1490 Nome: Redenção XIII | Bote 1 Reg. Capitania: 5452 Qtde Pessoas: 498 Inflável: Não | Bote 2 Reg. Capitania: 2346 Qtde Pessoas: 328 Inflável: Sim | Bote 3 Reg. Capitania: 438 Qtde Pessoas: 157 Inflável: Sim |
|--|--|--|---|

Crie um programa que implemente a estrutura de classes, bem como um programa principal que simule a utilização dessas classes considerando o exemplo de dados acima.

5. Uma empresa de eventos promove diferentes tipos de espetáculos. Para controlar os eventos e suas avaliações, está sendo desenvolvido um sistema com a seguinte estrutura de classes:



- A classe Evento é classe-base para a classe Show Musical, que é um tipo de evento promovido. Portanto, Show Musical herda as características de Evento;
- A classe Avaliação é parte da classe Evento (agregação)
- O método Mostrar é implementado tanto na classe-base quanto na descendente. Cada uma responde a este método mostrando todos os dados que possui (herdados ou próprios)
- A interface ao usuário questionará se o evento a ser cadastrado é Qualquer ou Show Musical, para então criar o objeto correspondente.

Ao final, imprimir todos os eventos cadastrados, com os dados da respectiva avaliação.

Exemplo de dados:

| |
|---|
| Evento: Teatro às 18h, no dia 10/09/2010, ingressos a R\$ 10,00, teve 1500 pagantes que acharam o evento Bom nível. |
| Evento: Palestra com João Silva, no dia 05/08/2010, ingressos a R\$ 2,00, teve 54 pagantes que acharam o evento Com informações esclarecedoras sobre as artes plásticas contemporâneo |
| Show Musical Tempo passado de rock progressivo do Grupo XYZ com ingressos a R\$ 15,00, contou com 265 pagantes no dia 15/09/2010. Opinião: Muuuito louco... |

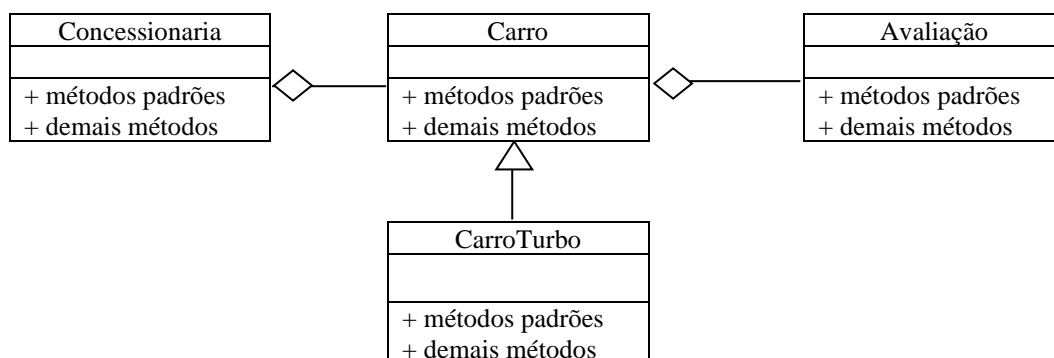
6. Uma concessionária de venda/compra de veículos usados necessidade de um sistema para controlar seus veículos. A concessionária precisa que para todo carro comprado pelo seu departamento de compras seja feito uma avaliação do estado mecânico do carro. Esta avaliação é uma listagem de itens mecânicos e um aval declarando a situação atual de tal item. Este aval pode assumir os seguintes valores: (necessita de troca, ruim, razoável, bom, ótimo).

A concessionária trabalha com tipos de carros: carro comum e carro turbo. O carro comum possui como atributos a placa, ano, potência do motor e tipo de combustível (1 – álcool e 2 – gasolina). Os métodos são: os construtores, um método para exibição dos atributos e outro de cálculo de rendimento do combustível (em Km/litro). O cálculo de rendimento do combustível deve seguir a seguinte fórmula:

- Para combustível do tipo 1: $\text{rendimento} = 10 / (\text{potencia} * 0,7)$
- Para combustível do tipo 2: $\text{rendimento} = 12 / (\text{potencia} * 0,6)$

Os carros do tipo turbo, além dos atributos de um carro comum, possuem uma taxa de potência adicional 30% em relação a potência do motor. Para esse tipo de carro, o cálculo de rendimento ficaria da seguinte forma:

- Para combustível do tipo 1: $\text{rendimento} = (10 / ((\text{potencia}+30\%) * 0,7))$
- Para combustível do tipo 2: $\text{rendimento} = (12 / ((\text{potencia}+30\%) * 0,6))$



Ciente dessas informações, implemente as classes descritas. Em seguida, implemente a classe concessionária que tem vários carros. O gerente da concessionária poderá consultar as informações de um determinado carro pela sua placa. Por fim, o sistema deverá fornecer um conjunto de cinco relatórios: relatório com a situação mecânica de todos os veículos para que os mesmos sejam enviados para uma oficina; o(s) carro(s) mais novo e o velho; o(s) carro(s) do tipo comum com maior rendimento; o(s) carros(s) do tipo turbo com menor rendimento e a média de rendimento deste tipo de carro; o(s) carro(s) com mais problemas mecânicos (aval = ruim ou razoável).