

## 2o. Trabalho de Programação Orientada a Objetos

**Data de Entrega:** 02/05/2024

**Natureza do Trabalho:** individual

**Forma de Entrega:** Pasta compactada do projeto gerado pela IDE NetBeans 8.2 utilizando a linguagem Java. A pasta do projeto deve ser nomeada da seguinte forma:

**prjPOOTrab2N** onde **N** representa o nome e sobrenome do aluno

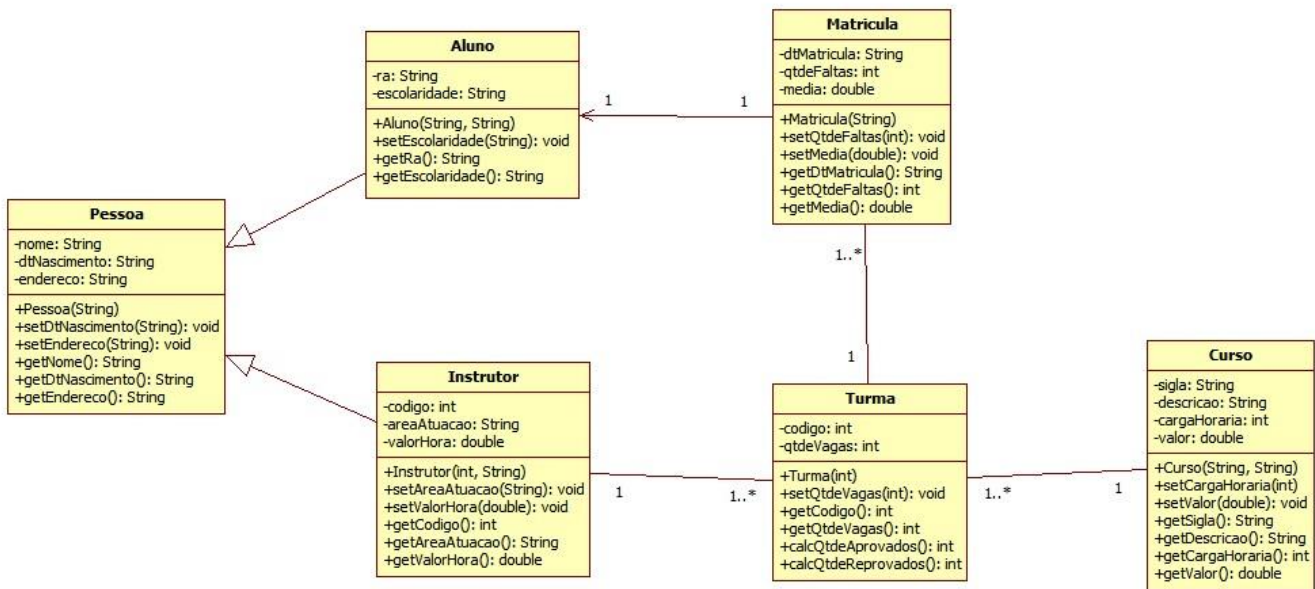
Exemplo: **prjPOOTrab2CarlosSilva**

- O trabalho deve ser enviado para o e-mail [dimas.cardoso@fatec.sp.gov.br](mailto:dimas.cardoso@fatec.sp.gov.br) com o seguinte título:

### **Entrega do 2º. Trabalho de POO**

- Na mensagem do e-mail deve constar o Ra e o Nome completo do aluno com a pasta compactada do projeto em anexo
- O trabalho deve ser enviado no horário das **08:00 às 22:00**
- Trabalho enviado fora do horário especificado será desconsiderado

A partir do diagrama de classes abaixo faça a implementação, na linguagem java, de cada classe com as suas respectivas interações (generalização/especialização e associação binária). Ao final, faça a implementação da classe Aplic de acordo com as orientações definidas neste trabalho.



## Especificações

- Método Pessoa tem como parâmetro de entrada o nome.
- Método Instrutor tem como parâmetros de entrada o código e o nome.
- Método setAreaAtuacao tem como parâmetro de entrada a área de atuação do Instrutor (Ex.: Desenvolvimento Web, Banco de Dados, Redes).
- Método Aluno tem como parâmetros de entrada o ra e o nome.
- Método setEscolaridade tem como parâmetro de entrada a escolaridade (Ex: Ensino Médio, Ensino Superior, PosGraduação).
- Método Matricula tem como parâmetro de entrada a data de matricula.
- Método Turma tem como parâmetro de entrada o codigo.
- Método calcQtdeAprovados calcula e devolve a quantidade de alunos aprovados. O aluno será considerado aprovado caso tenha uma frequência mínima de 60% da carga horária do curso e uma média com valor igual ou superior a 5,0.
- Método calcQtdeReprovados calcula e devolve a quantidade de alunos reprovados.
- Método Curso tem como parâmetros de entrada a sigla e a descrição.
- Método setCargaHoraria tem como parâmetro de entrada a quantidade de horas do curso.
- Método setValor tem como parâmetro de entrada o valor do curso.

## Orientações

- A identifique a classe e em seguida implemente na classe identificada o método **+listarAlunos():void**. Este método deve produzir a seguinte saída de dados:

Relação Alunos da Turma			
Sigla do Curso: xxxxxxxxxxxx		Nome do Curso: xxxxxxxxxxxxxxxxxxxx	
Sigla Turma: xxxxxxxx			
Nome do Instrutor: xxxxxxxxxxxxxxxx			
Nome do Aluno	Qtde. Faltas	Média	Situação
xx	99	99,9	xxxxxxxxxxxxxxxx
xx	99	99,9	xxxxxxxxxxxxxxxx
xx	99	99,9	xxxxxxxxxxxxxxxx
.	.	.	.
.	.	.	.
.	.	.	.
Qtde. de Alunos: 99			
Qtde. de Alunos Aprovados : 99			
Qtde. de Alunos Reprovados: 99			

A coluna **Situação** indica a situação do aluno no curso (Aprovado ou Reprovado). O aluno será considerado aprovado caso tenha uma frequência mínima de 60% da carga horária do curso e uma média com valor igual ou superior a 5,0. Utilize o método **listarAlunos** na classe a **Aplic**

- Utilize o **this** para simplificar a utilização da associação binária bidirecional.
- Utilize a classe **ArrayList** para fazer a implementação da multiplicidade **1..\***.
- Na classe **Aplic** utilize a classe Scanner para fazer a entrada de dados.
- Na classe **Aplic** faça a instanciação de:
  - 1 objeto da classe Curso
  - 2 objetos da classe Turma
  - 1 objeto da classe Instrutor
  - 5 objetos da classe Aluno e da classe Matricula
- Na classe **Aplic** faça a interação entre os objetos instanciados de modo que os três primeiros objetos da classe Aluno devem interagir com o primeiro objeto da classe turma e os dois últimos objetos da Aluno devem interagir com o segundo objeto da classe Turma.