

# Introdução a UML

# Introdução a UML

Neste ponto surge a questão:

Como modelar os sistemas desenvolvidos com orientação a objeto?

Utilizando um método de modelagem.

Qual método utilizar?

Na década de 1980 esta resposta era muito complexa. Pois, existiam inúmeros métodos, cada um com sua notação para representar o mesmo conceito.

## Introdução a UML

Esta situação gerava um crescente problema, devido ao grande avanço na utilização da orientação a objeto (OO) no desenvolvimento de sistemas.

Percebendo isto, em 1994 dois conceituados metodologistas (James Rumbaugh e Grady Booch) efetuaram a unificação de seus métodos. Buscando minimizar os transtornos gerados pela ausência de uma metodologia padrão para OO.

O método foi publicado sob o título Unified Method 0.8.

No ano de 1995 outro metodologista, Ivar Jacobson, se juntou à dupla, gerando em outubro de 1996 a versão 0.91 da Unified Modeling Language ou apenas UML.

## Introdução a UML

Desde então a UML vem se consolidando cada vez mais como linguagem padrão para modelagem de sistemas orientados a objeto.

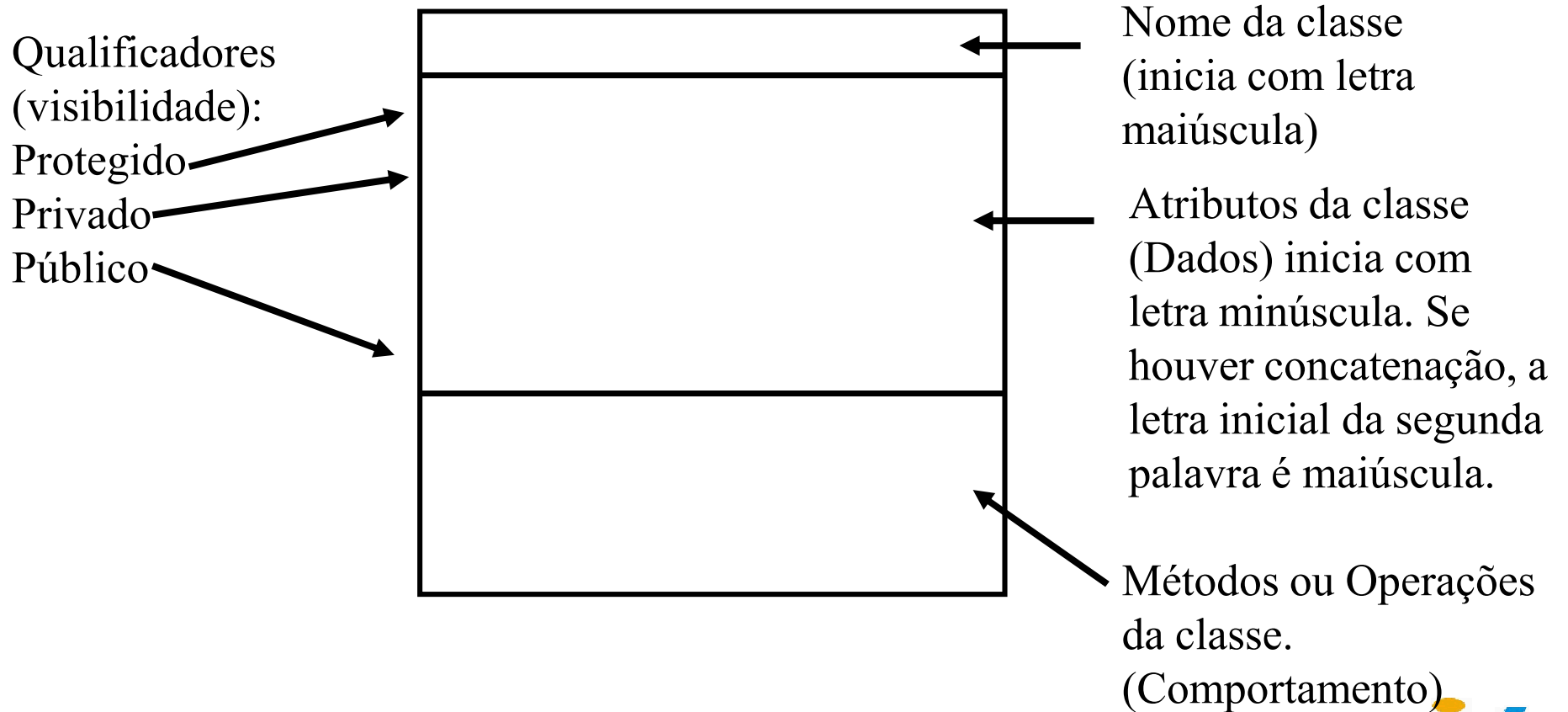
Numa definição mais geral UML é uma linguagem para especificação, construção, visualização e documentação de sistemas de software. Representando a união da sintaxe gráfica de vários métodos, com vários símbolos removidos e vários adicionados.

Detalharemos agora o **diagrama de classes em UML**.

# Introdução a UML

Uma classe em UML é representada por um retângulo dividido em três partes.

Por exemplo:



## Introdução a UML

Existem diversos softwares para gerar diagramas em UML.

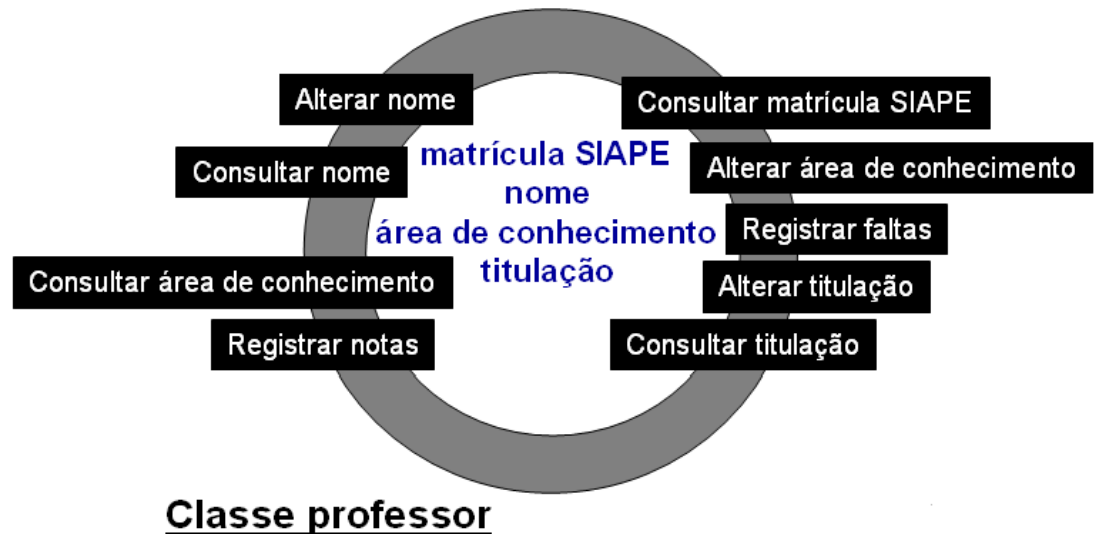
Trabalharemos com o **Dia**, pois este serve para criar inúmeros tipos de diagramas, como por exemplo, diagramas UML, diagramas de fluxos, diagramas de rede e diagramas elétricos.

Duas características do Dia influenciaram em sua escolha, o fato deste ter versões para Linux e para Windows além de ser um software gratuito.

# Introdução a UML

**Exercício:** Represente a classe professor, discutida anteriormente, em UML.

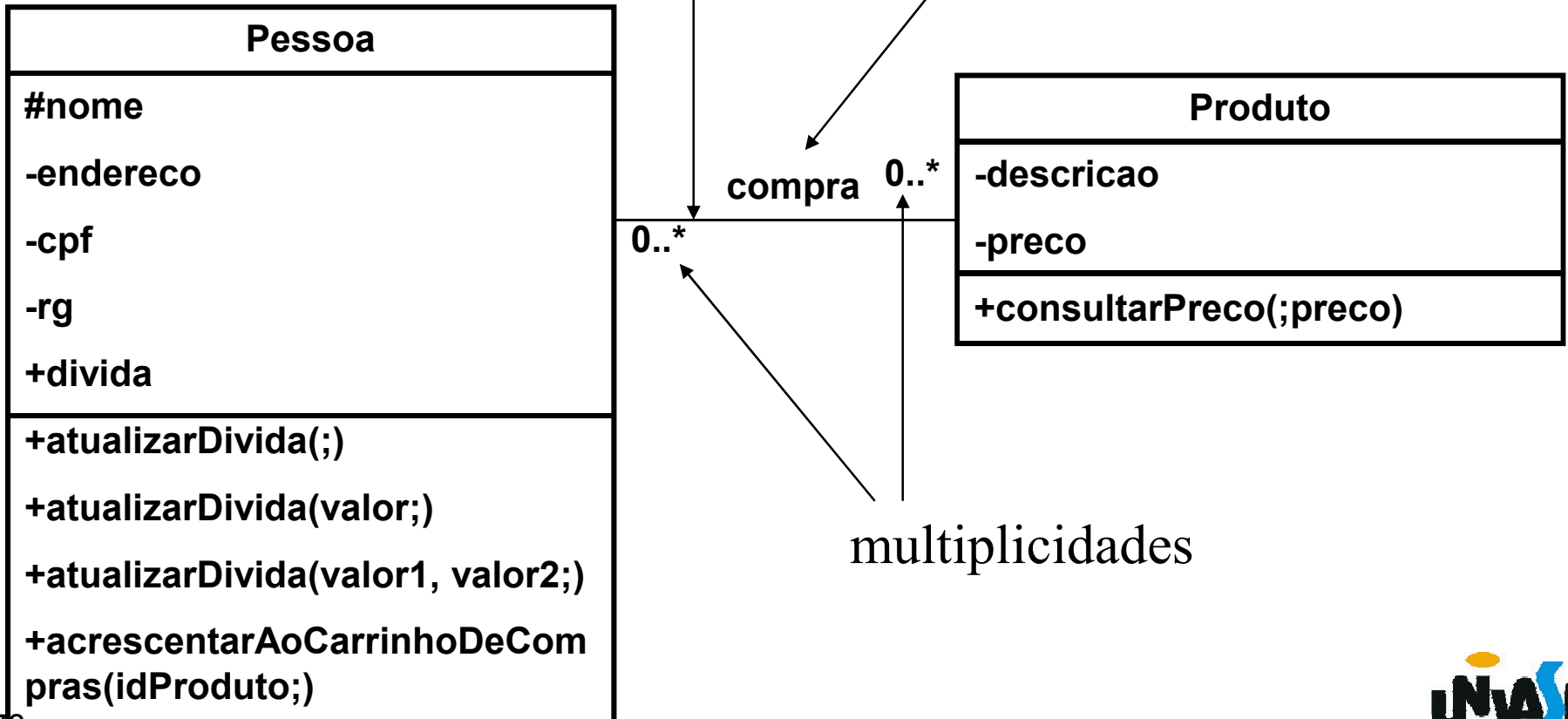
Professor
-matriculaSIAPE
-nome
-areaDeConhecimento
-titulacao
+alteraNome (nome;)
+consultaNome(;nome)
+consultaAreaDeConhec (;area)
+registraNotas(turma;)
+consultaMatSIAPE(;mat)
+alterarAreaDeConhec(area;)
+registraFaltas(turma;)
+alteraTitulacao(titulacao;)
+consultaTitulacao(;titulacao)



# Introdução a UML

Uma associação em UML é representada por uma linha ligando os retângulos que representam as classes envolvidas.

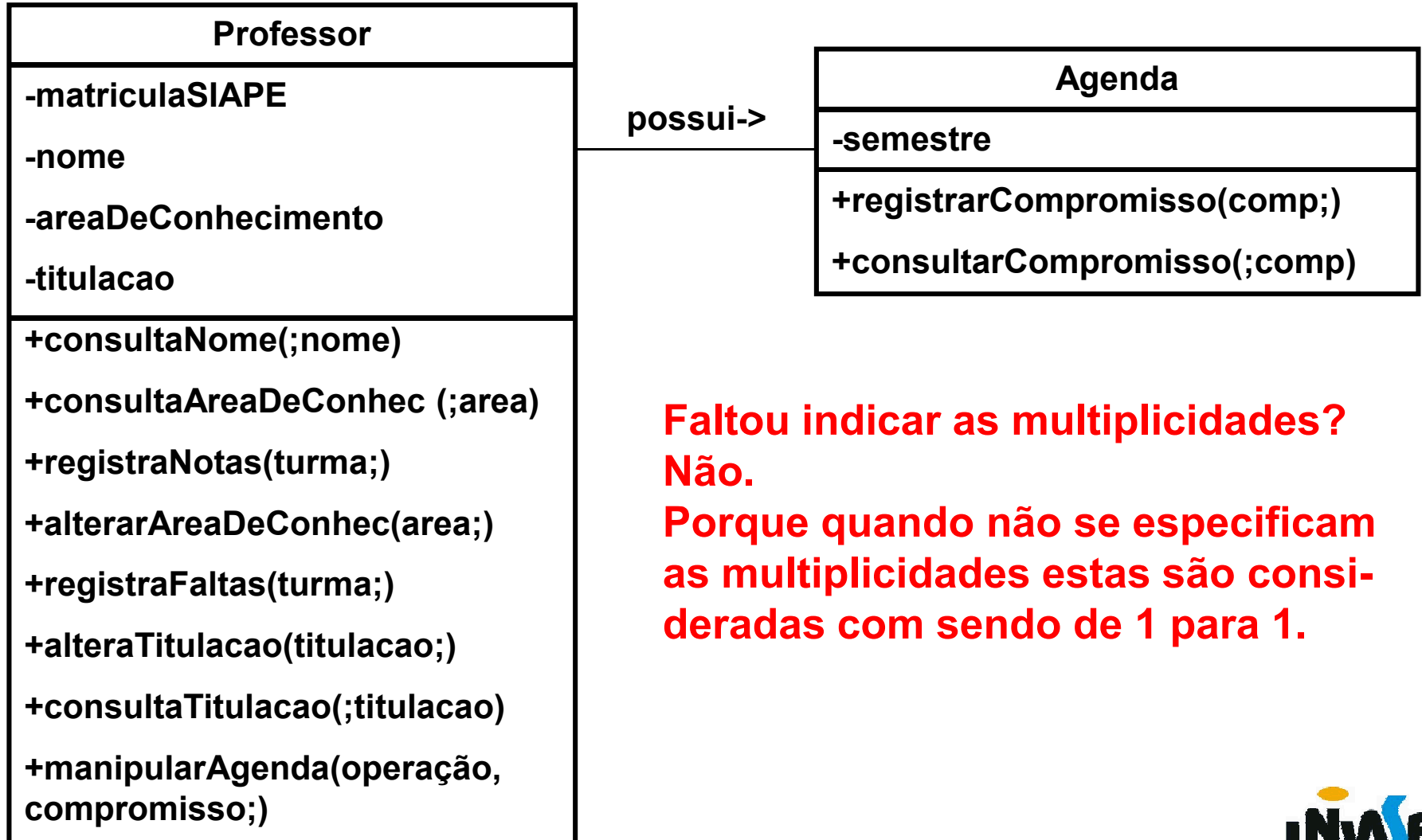
Por exemplo:






# Introdução a UML

**Exercício:** Represente uma associação, utilizando UML, considerando um sistema de gestão acadêmica.

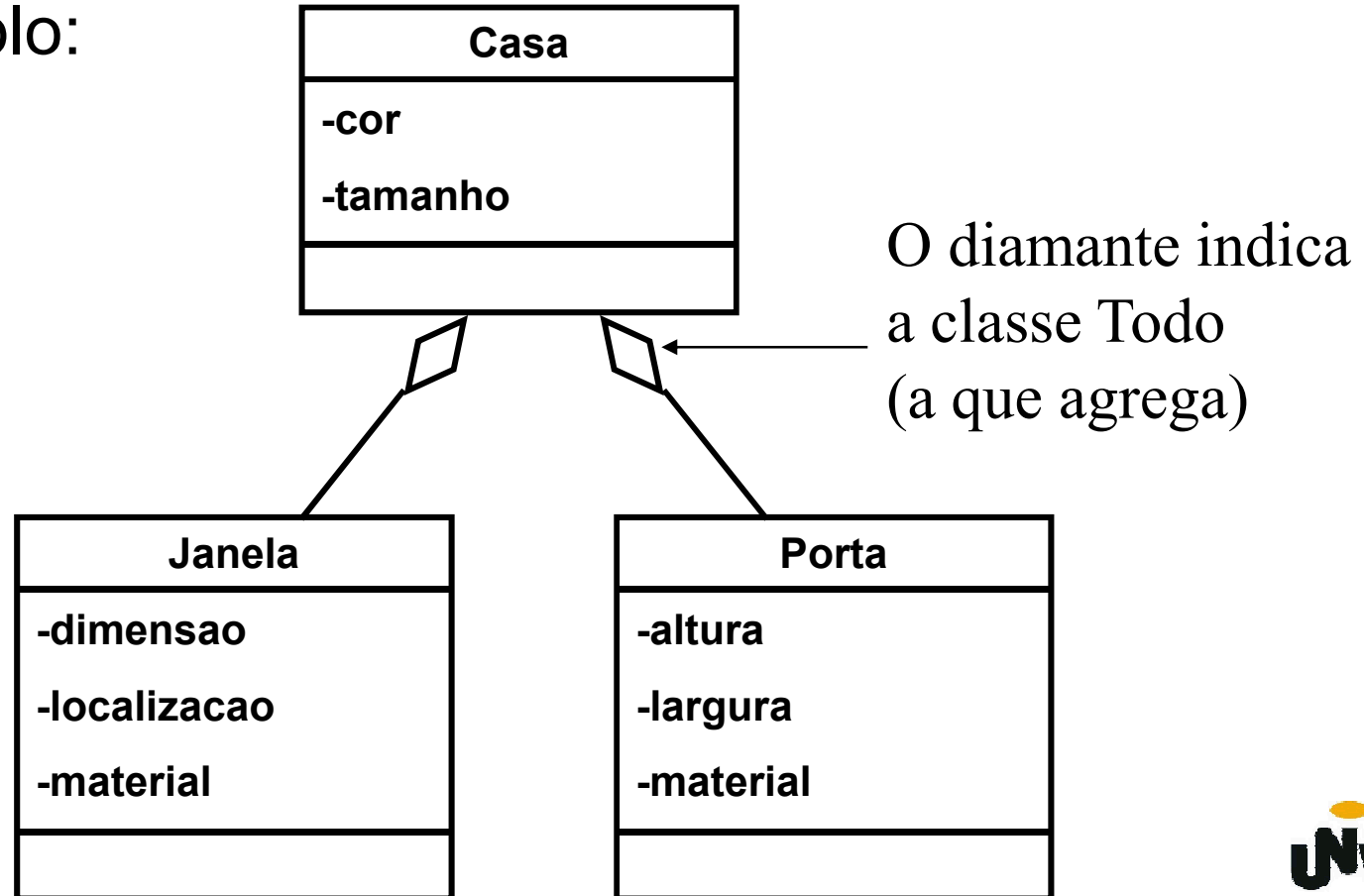


**Faltou indicar as multiplicidades?  
Não.  
Porque quando não se especificam  
as multiplicidades estas são consi-  
deradas com sendo de 1 para 1.**

# Introdução a UML

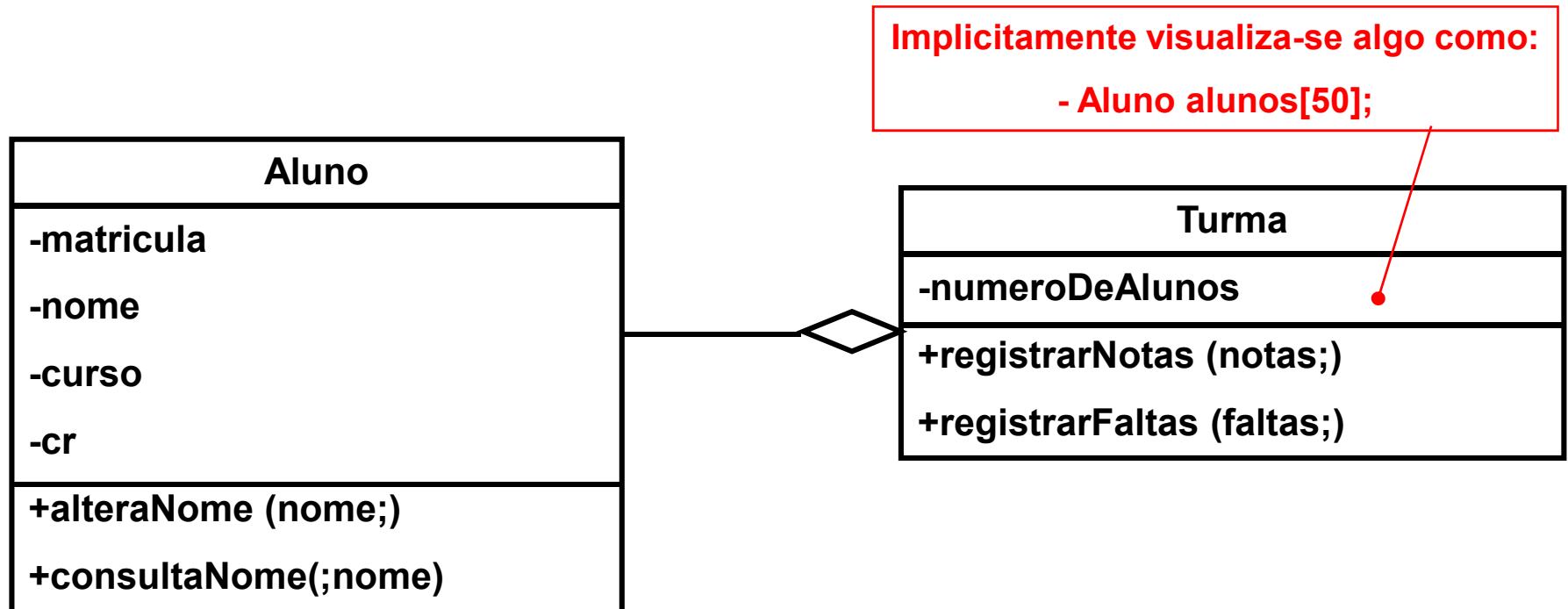
Uma agregação em UML é representada por  ligando os retângulos que representam as classes envolvidas.

Por exemplo:



# Introdução a UML

**Exercício:** Represente uma agregação, utilizando UML, considerando um sistema de gestão acadêmica.

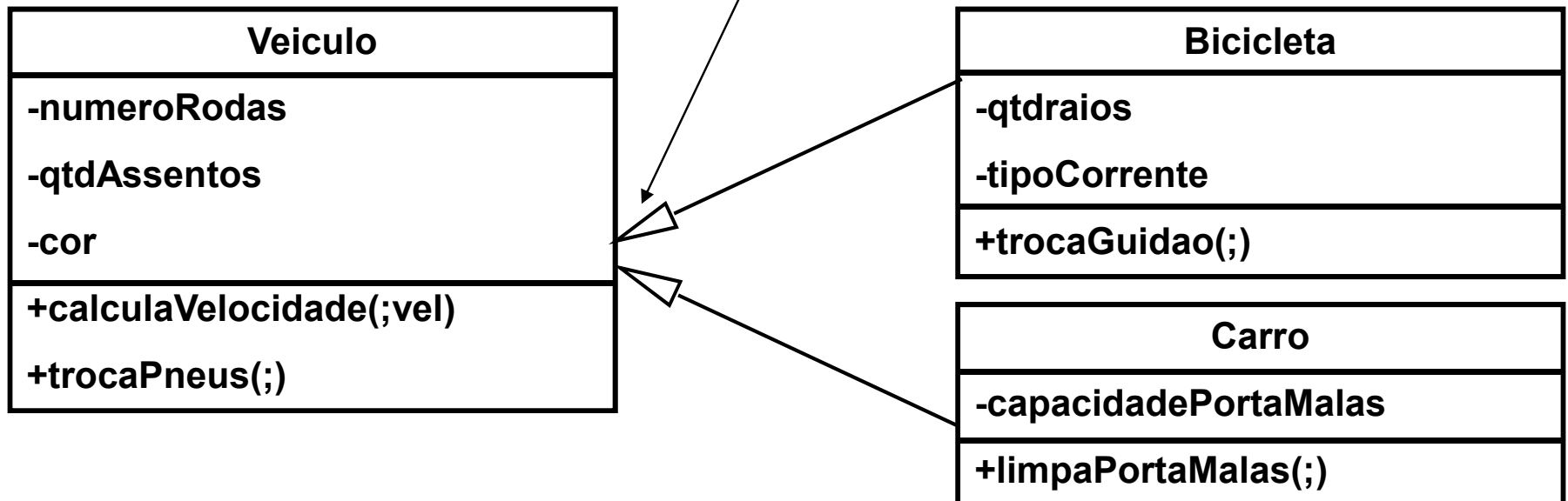


# Introdução a UML

Uma generalização (herança) em UML é representada por uma seta ligando os retângulos que representam as classes envolvidas.

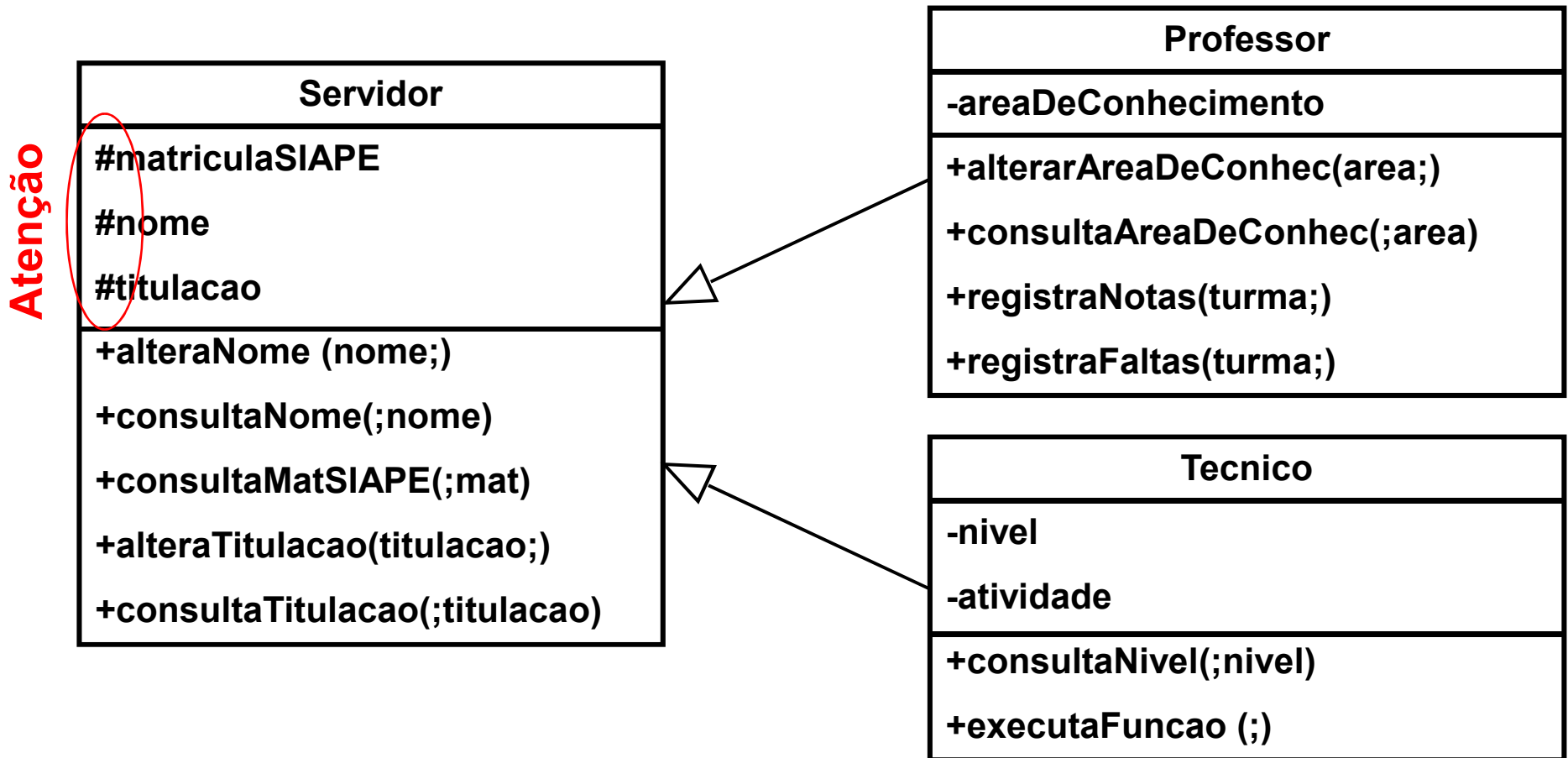
Por exemplo:

A seta indica a superclasse



# Introdução a UML

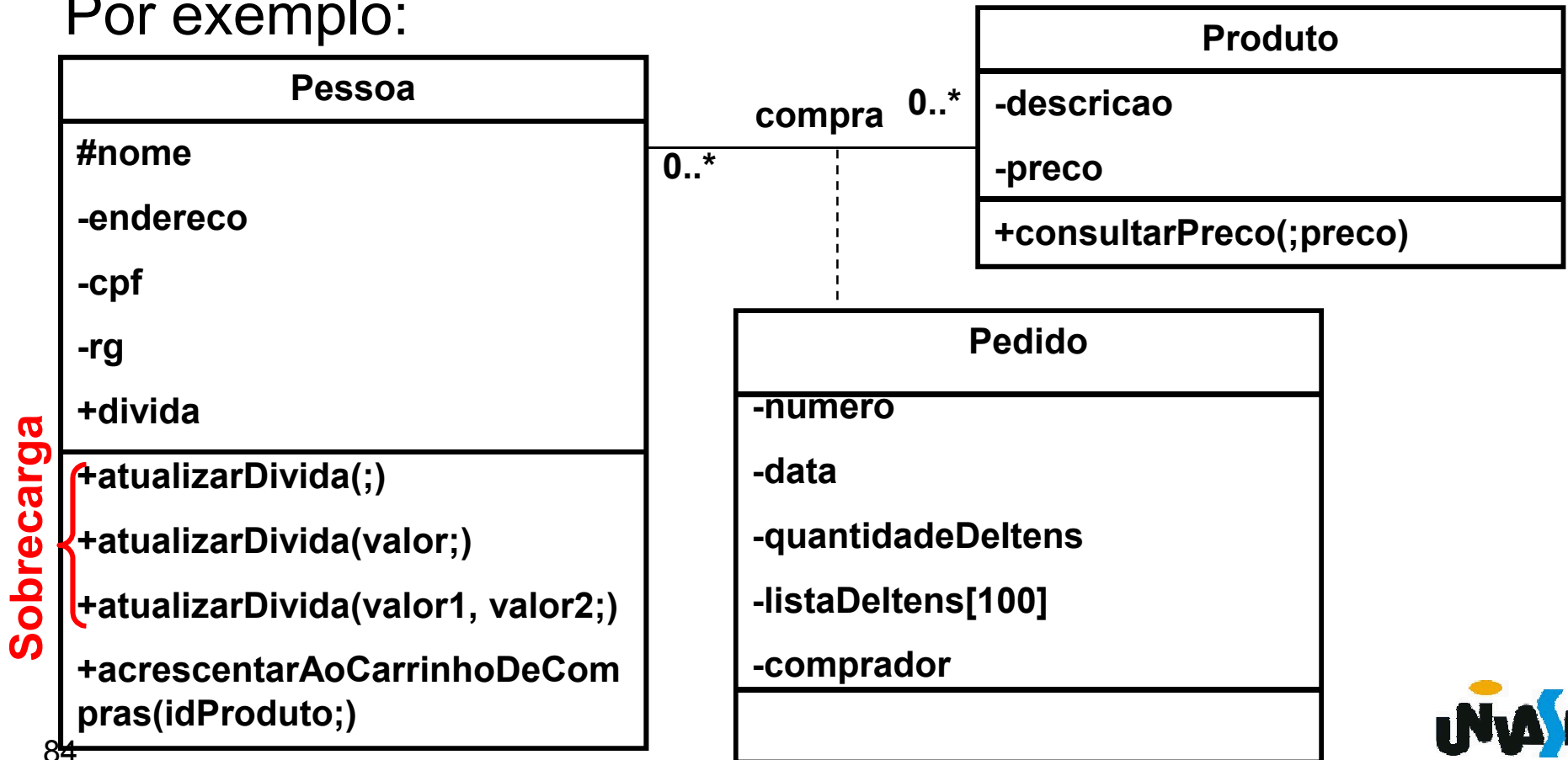
**Exercício:** Represente uma generalização, utilizando UML, considerando um sistema de gestão acadêmica.



# Introdução a UML

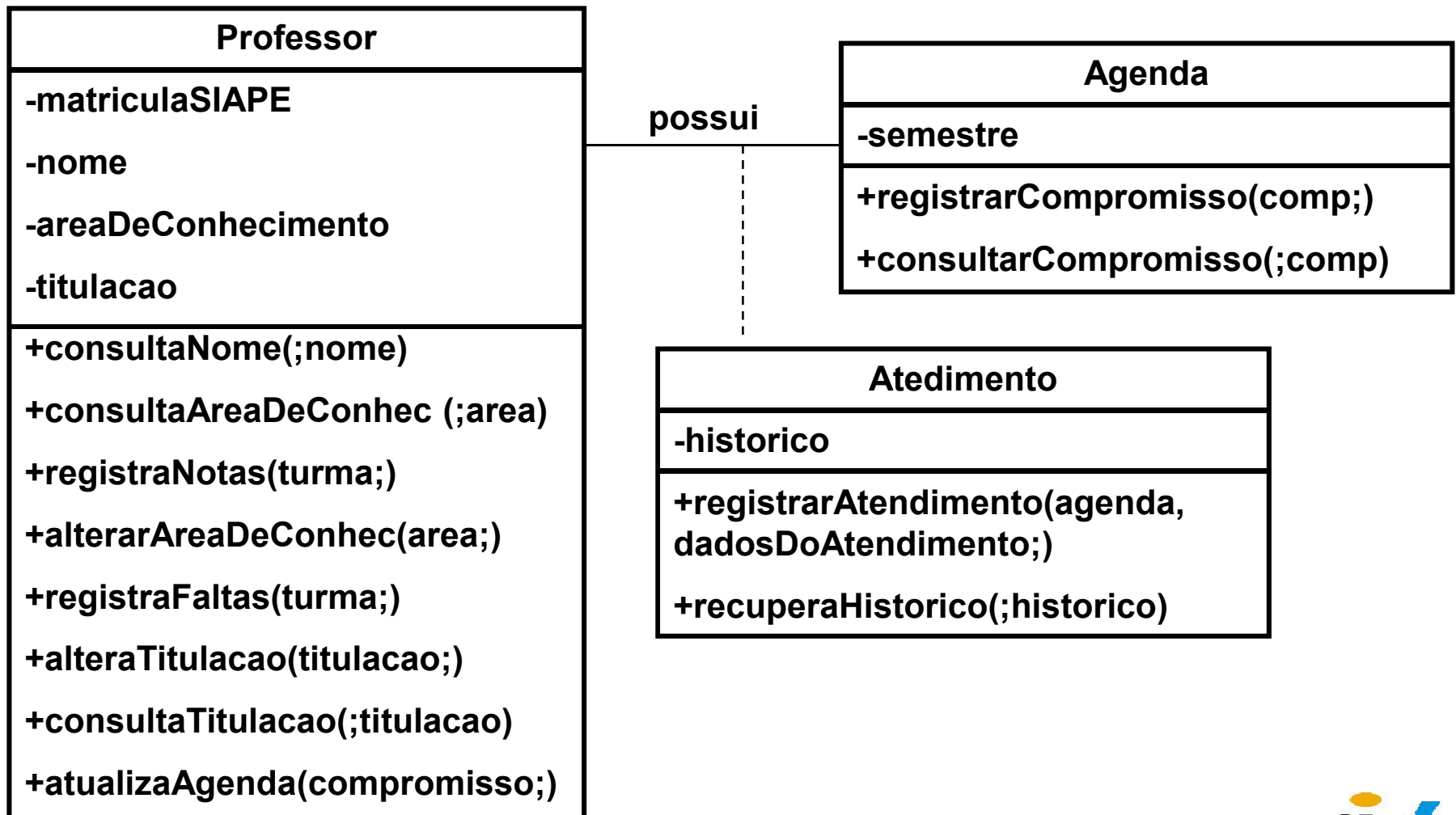
O relacionamento de uma classe com uma associação em UML é representada por uma linha tracejada ligando a linha da associação ao retângulo que representa a classe em questão.

Por exemplo:



# Introdução a UML

**Exercício:** Represente relacionamento de uma classe com uma associação, utilizando UML, considerando um sistema de gestão acadêmica.



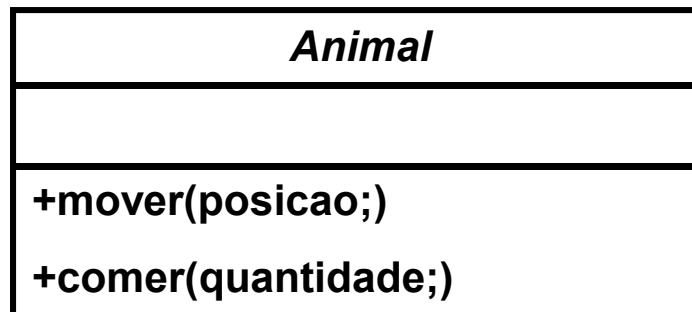
# Introdução a UML

## Classe abstrata

Uma classe abstrata é uma classe que não pode ser instanciada, sendo assim, sua finalidade fica restrita a definir características (atributos) e comportamentos (métodos) para que outras classes possam herdar.

Em UML representamos uma classe abstrata colocando seu nome em *itálico*.

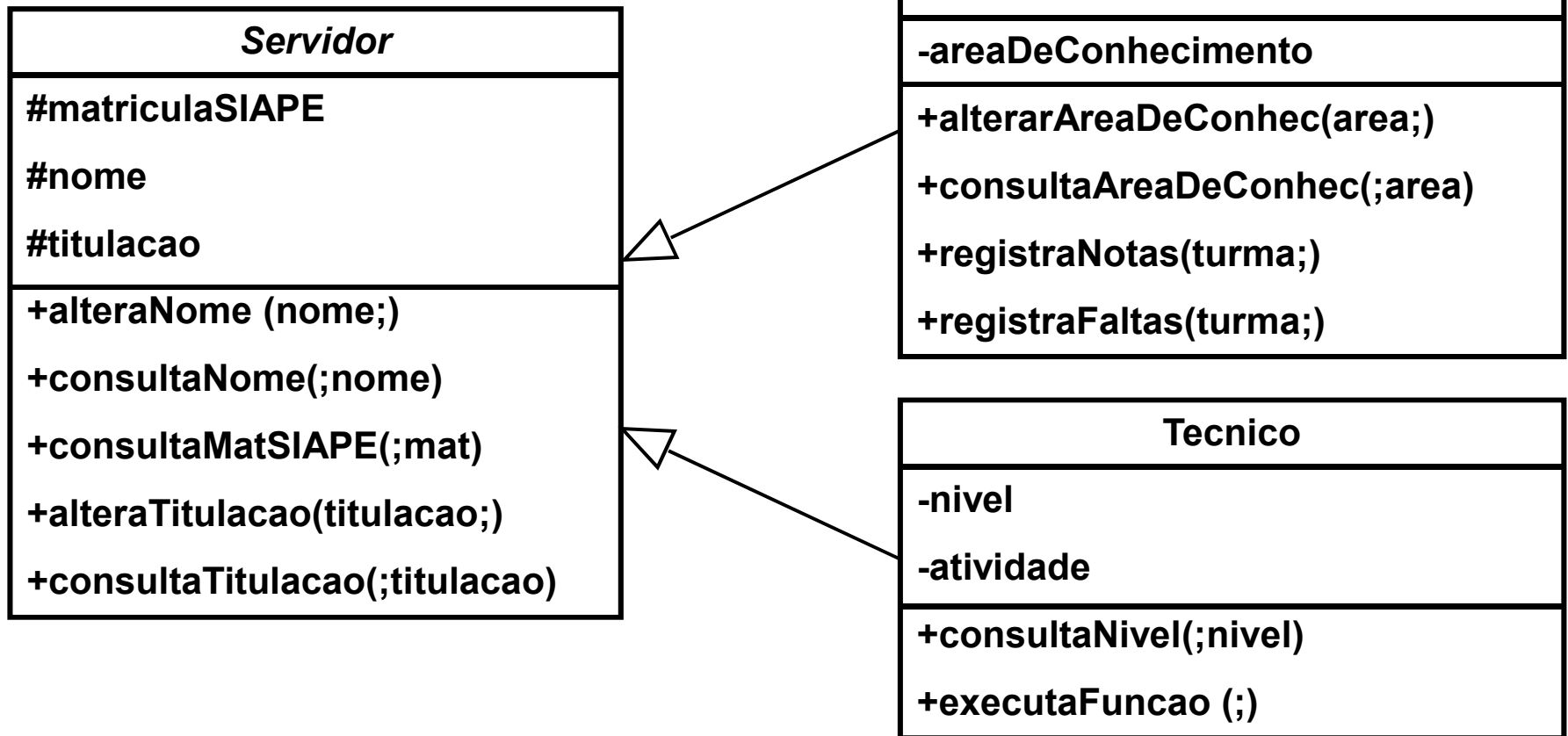
Por exemplo:





# Introdução a UML

**Exercício:** Represente uma classe abstrata, utilizando UML, considerando um sistema de gestão acadêmica.



# Introdução a UML

## Exercício:

Com base nos conceitos estudados, modele, utilizando UML, um sistema OO para gerenciar o agendamento de consultas em uma clínica odontológica, considerando a realização de serviços. Tente explorar ao máximo os tópicos trabalhados.