Linguagens de Programação

Noções POO

José Martins
Escola Superior de Tecnologia
Instituto Politécnico do Cávado e do Ave
jmartins@ipca.pt

Adaptado de: António Araújo

Programação Orientada a Objetos

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

Programação Orientada a Objectos:

- Paradigma de Programação focada no conceito de "objeto"
- Paradigma de Programação:
  - Visão que o programador tem …
- Objeto
  - Entidade representativa de um problema
- O programador pensa em termos de objetos ao avaliar um problema

Programação Orientada a Objetos

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- A origem deste paradigma vem de 2 áreas
  - Simulação (anos 60)
  - Engenharia de Software (anos 70)
- Começa a ter impacto na tecnologia nos anos 90
- É o paradigma de programação mais difundido
  - Exemplos: C++, C#, Java, Python, PHP

Simulação

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- O ramo da simulação pretendia encontrar formas de modelar em computador, entidades do mundo real que possuem uma estrutura interna (atributos próprios) e cujo comportamento se quer analisar.
- Estas entidades são consideradas moldáveis se possuírem:
  - Identidade (única)
  - Estrutura (propriedades definidas por atributos)
  - Comportamento (ações e reações)
  - Interação (relação com outras entidades)

Simulação

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

objeto - entidade abstrata, modelo de entidade real

- Exemplo: operário, carro, turma, aluno, professor, disciplina, …

classe - entidade geradora e agregadora de todos os objetos com um padrão comum de estrutura e comportamento

Simulação

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

Vivemos num mundo de objetos:

- feitos pelo homem
- de negócio
- produtos que usamos
- …

Estes objetos podem ser:

- criados, categorizados, descritos, organizados, combinados, manipulados, …

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

Objetivos da Engenharia de Software:

- Diminuição da complexidade dos projetos, dos tempos de desenvolvimento, dos custos e dos riscos associados
  - Fazer mais em menos tempo e com menos dinheiro

- criação de unidades de programação (componentes) que permitam
  - maior abstração -> mais fácil programar
  - maior reutilização -> menos trabalho e maior segurança

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

Evolução deste ramo vêm de duas tendências:

- modelação e programação centrada nos processos (instruções)
  - C, Pascal, …
- modelação e programação centrada nos dados
  - Bases de dados relacionais, …

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

Esta evolução foi condicionada pelas limitações do hardware:

- O armazenamento (memória, discos, …) era caro
- Os computadores eram muito grandes e com baixa capacidade de processamento

- Por causa desta limitação havia uma grande necessidade em reutilizar Software de modo a evitar que um projeto comece do "zero".
  - Desenvolver código que possa ser aproveitado noutros projetos

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Programa - sequência de instruções que operam sobre dados
- Antes os programas tinham um controlo de execução limitado
- Mas introduziram-se estruturas de controlo:
  - ciclos, funções e procedimentos
  - e assim, já foi possível implementar formas simples de reutilização

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- A estrutura de um programa que antes era monolítica (lista de comandos a executar pelo computador)

- Após a criação destas estruturas, um programa passou a ser composto por um programa principal e por um conjunto de funções e procedimentos que são utilizáveis pelo programa principal através de **invocações**

Engenharia de Software

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Modelos de Bases de Dados Relacionais
  - Clara separação entre os dados e os procedimentos que os manipulam
  - modelação e programação centrada nos dados
- Forma de pensar, que ainda hoje é maioritariamente usada nas bases de dados consiste em:
  - Pensar nos dados
  - Pensar nas instruções que vão manipular os dados

Noção de Objeto

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

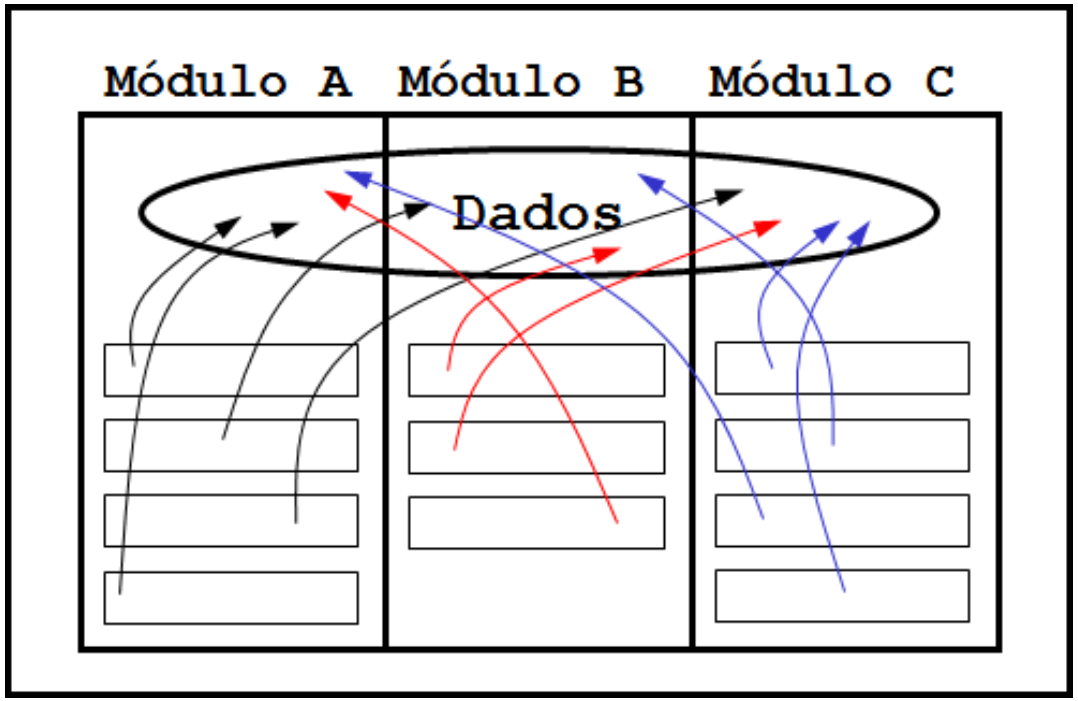
- Unidades de computação que contêm declarações de dados juntamente com declarações de procedimentos e funções
- Possuem a propriedade de poderem ser compilados isoladamente e serem ligados posteriormente a qualquer programa que deles necessite
- Linguagens: PASCAL, C
- Características Fundamentais:
  - Abstração de controlo ou de funcionalidade
  - Reutilização de código

Noção de Objeto

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- No início haviam limitações pois os dados poderiam ser globais a vários módulos e não eram completamente autónomos

•Dependiam do contexto em que eram utilizados



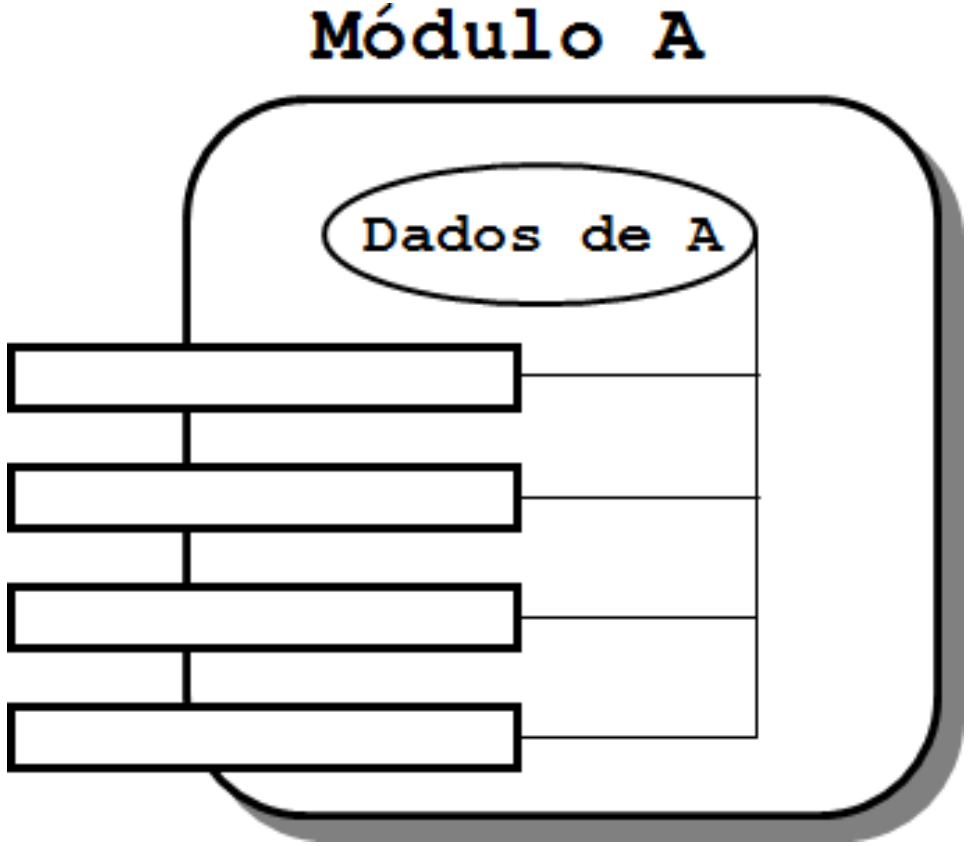
Noção de Objeto

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Mas depois surgiram módulos independentes do contexto:
  - utilizáveis em qualquer programa e absolutamente autónomos
  - os seus procedimentos apenas podem aceder a variáveis locais ao módulo
  - no código dos seus procedimentos não podem existir instruções de input e output

Noção de Objeto

- Os módulos contêm:
  - uma estrutura interna
  - funções e procedimentos:
    - é o único código com acesso às suas variáveis internas;
    - não podem aceder a outras variáveis que não sejam locais ao módulo



Módulos

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Os módulos passam a ser vistos como mecanismos de abstração de dados e não de abstração de controlo:
  - entidade fundamental: estrutura de dados não disponível para o exterior
  - os procedimentos e funções passam a ser vistos como serviços para o exterior, para que do exterior se possa aceder à estrutura de dados

Noção de Objeto

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Baseia-se nas seguintes propriedades fundamentais:
  - independência de contexto -> reutilização
  - abstração de dados -> abstração
  - encapsulamento -> abstração e proteção
  - modulariedade -> desenvolvimento feito por composição de partes simples
- Um objeto é o módulo computacional básico e único que corresponde à representação abstrata de uma entidade autónoma

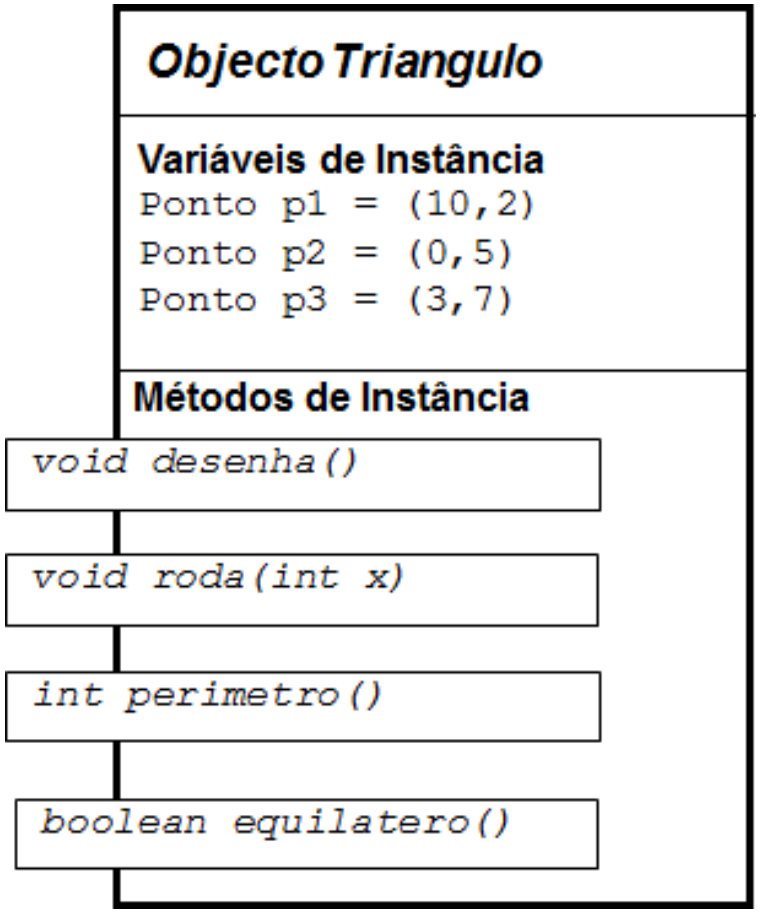
Noção de Objeto

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Um objeto é constituído por:
  - um identificador único
  - um conjunto privado de atributos

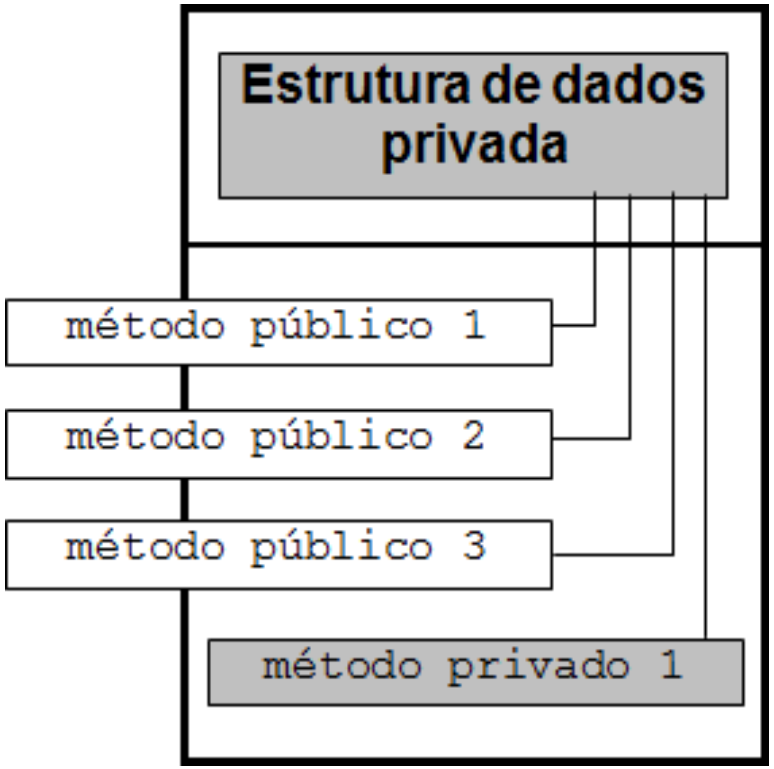
    variáveis de instância
  - um conjunto de operações

    métodos de instância



Encapsulamento

- Um objeto é uma cápsula que contém dados e métodos. Os dados apenas podem ser manipulados pelos seus métodos privados)
- Os seus métodos privados, tal como os dados, estão fechados dentro da mesma cápsula, não sendo visíveis do exterior



Mensagens

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

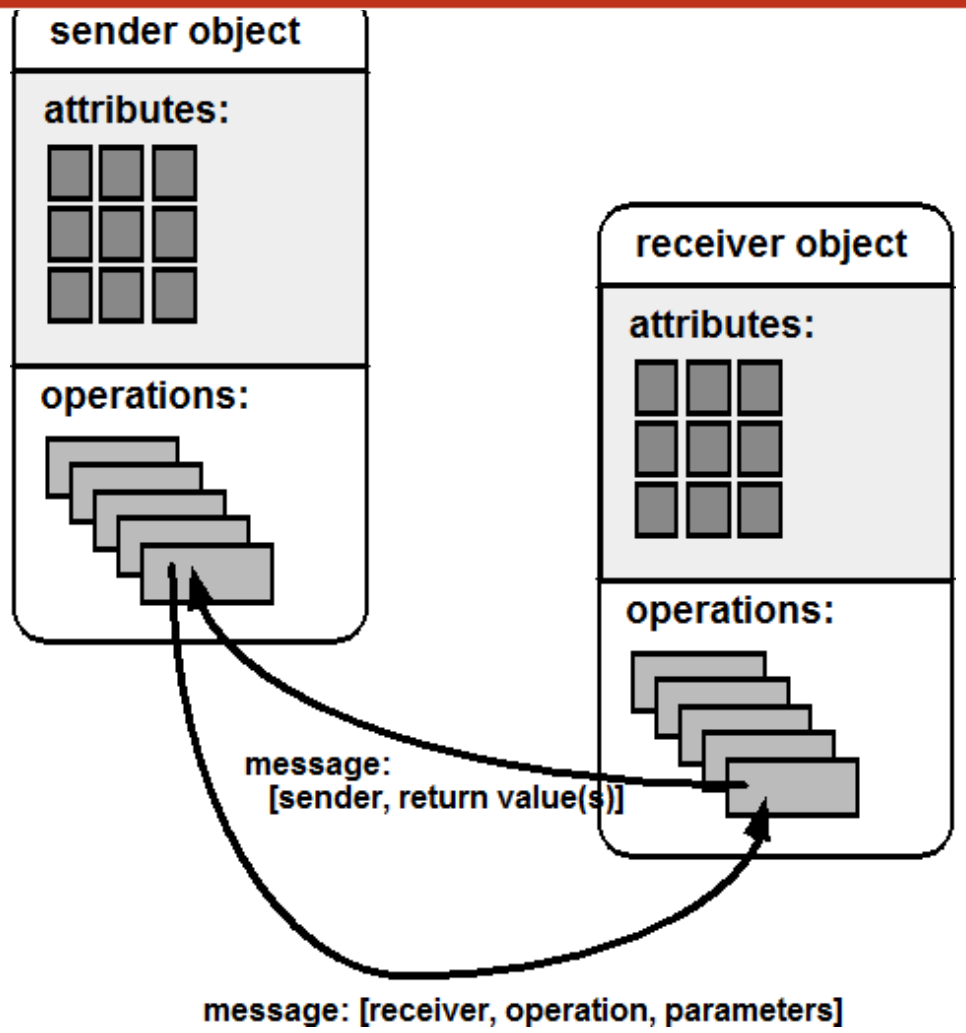
- Os objetos interagem entre si através de um mecanismo de envio e receção de mensagens
- Uma mensagem é:
  - uma invocação de um método público do objeto recetor da mensagem
  - o emissor passa o controlo de execução ao recetor da mensagem
  - o resultado da execução do método é devolvido ao emissor da mensagem
  - o recetor devolve o controlo de execução ao emissor da mensagem.

Mensagens

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Comunicação por mensagens
  - através de mensagens um objeto ativa métodos, envia dados e recebe dados de outros objetos
  - o envio de uma mensagem implica a transferência de controlo
- Sequência:
  - O emissor envia mensagem ao recetor
  - o recetor executa o método associado à mensagem
  - quando termina a execução do método, o recetor devolve o resultado e o controlo ao emissor

Mensagens


sender object
attributes:
operations:
receiver object
attributes:
operations:
message:
[sender, return value(s)]
message: [receiver, operation, parameters]

Objetos, Instâncias e classes

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- **Classe** - entidade que pode ser vista sobre duas perspetivas
  - Operacional:
    - Descreve - contém a descrição da estrutura e comportamento de objetos da mesma "família"
    - Constrói - cria os objetos
  - Semântica:
    - forma de classificar objetos

Objetos, Instâncias e classes

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- **Instância** são os objetos criados pelas classes
  - um objeto é instância de uma única classe
  - um objeto apenas pode ser criado por uma única classe
  - uma classe pode ter (criar) várias instâncias
  - instâncias da mesma classe
    - têm igual estrutura e comportamento
    - diferem nos valores que estão nas variáveis de instância

Programa

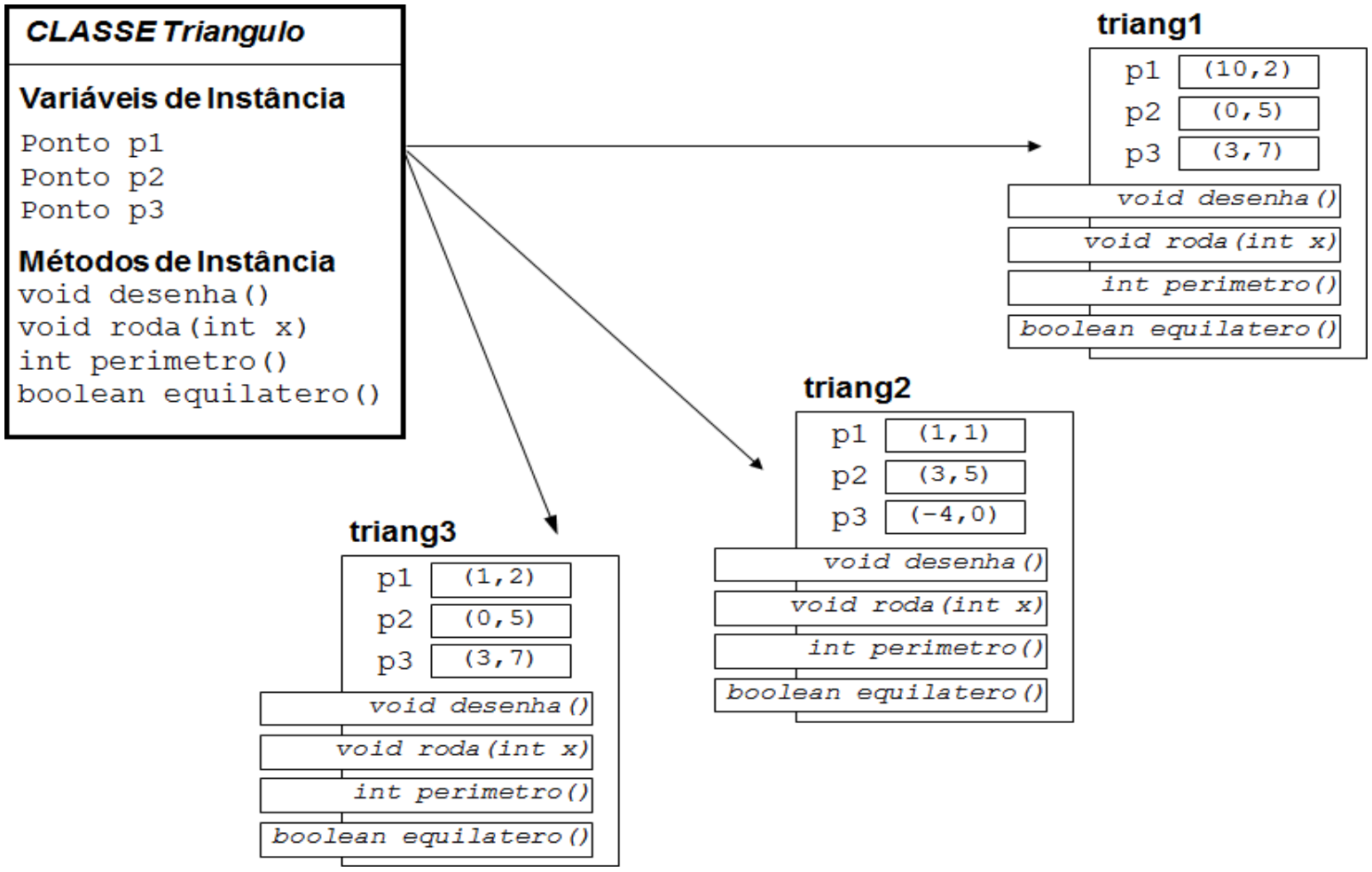
ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Um **objeto** tem existência, i.e., capacidade de execução, e interage com outros objetos através do envio e receção de mensagens
- Não existem dois objetos absolutamente iguais, diferem pelo menos na identificação
  - As entidades do mundo real são moldáveis se possuírem identidade (única)

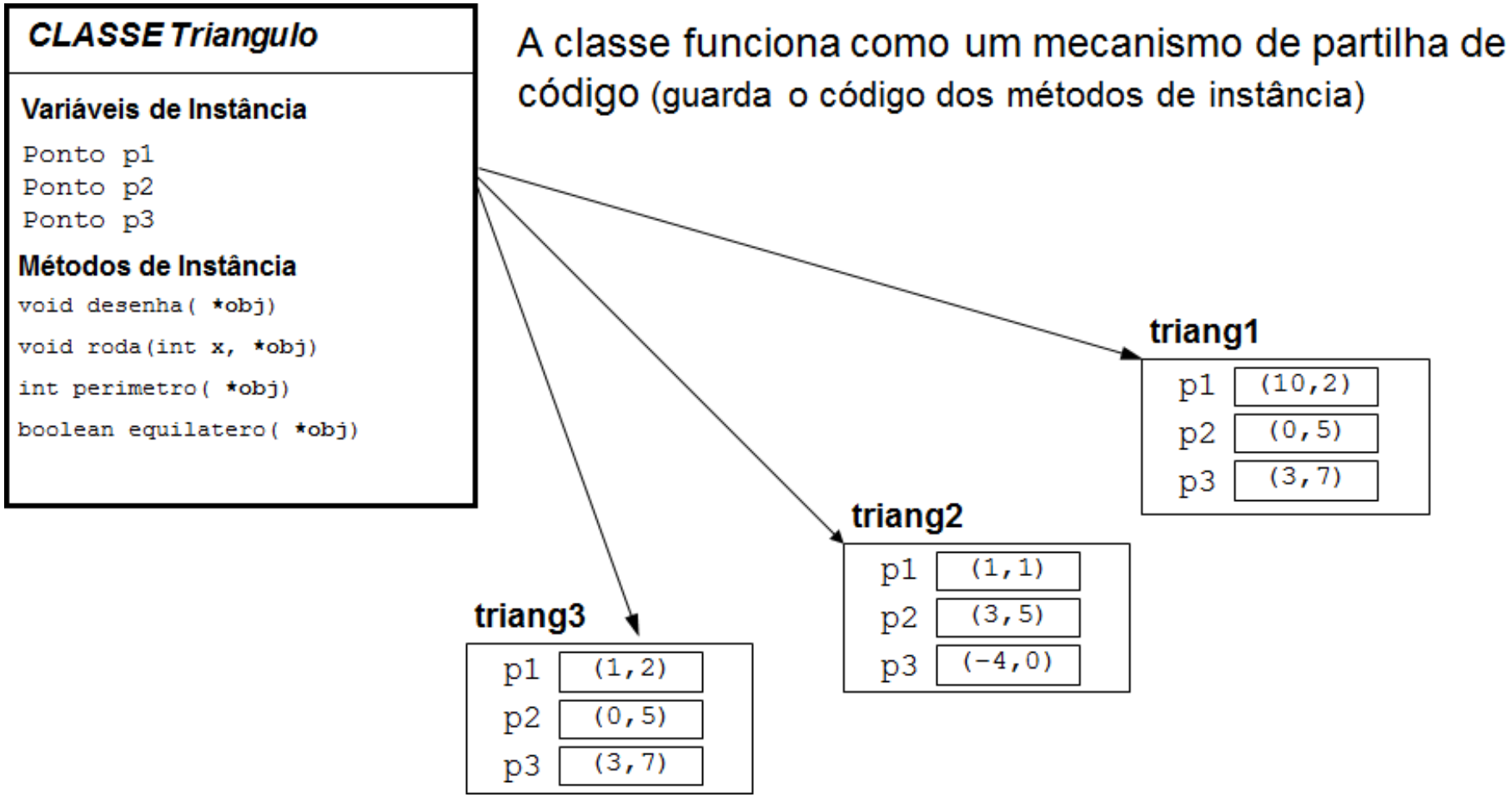
Objetos, Instâncias e classes

- **Instanciação** - processo que cria um objeto a partir de uma classe
- **Instância** - objeto resultante do processo de instanciação

Programa


CLASSE Triangulo
Variáveis de Instância
Ponto p1
Ponto p2
Ponto p3
Métodos de Instância
void desenha()
void roda(int x)
int perimetro()
boolean equilatero()
triang1
p1 (10,2)
p2 (0,5)
p3 (3,7)
void desenha()
void roda(int x)
int perimetro()
boolean equilatero()
triang2
p1 (1,1)
p2 (3,5)
p3 (-4,0)
void desenha()
void roda(int x)
int perimetro()
boolean equilatero()
triang3
p1 (1,2)
p2 (0,5)
p3 (3,7)
void desenha()
void roda(int x)
int perimetro()
boolean equilatero()

Objetos, Instâncias e classes



Programa

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Um programa em POO é o resultado da computação gerada pela interação (troca de mensagens) entre os vários objetos que o constituem.
- Um programa OO é constituído apenas por objetos, que por sua vez são criados a partir de classes.
- Um programa OO é construído com base:
  - nas classes que irão criar os objetos que o constituem
  - numa mensagem inicial, a ser enviada a uma classe, que irá desencadear todo o processo de construção dos objetos que o constituem.

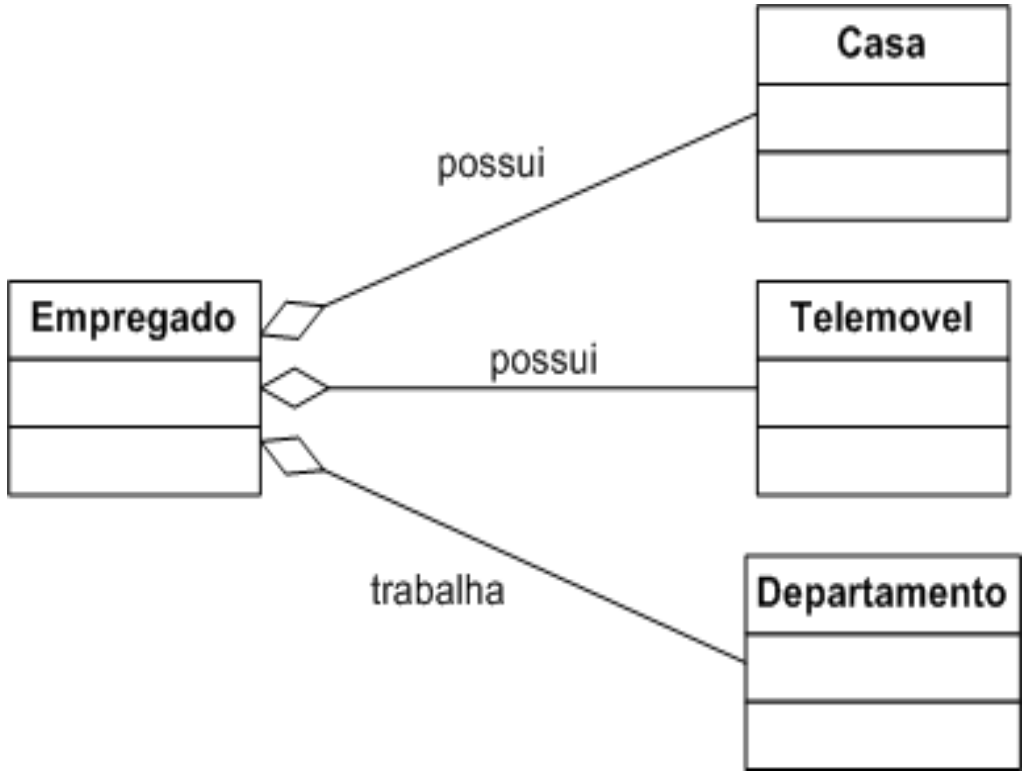
Programa

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- Como fazer um programa numa linguagem OO
  - Definir as classes que vão criar os objetos e os seus relacionamentos
    - Análise OO - Pensar em termos de objetos
    - Notação - Formal: UML
  - Definir um método especial que vai desencadear todo o processo de criação dos objetos num processo de explosão.

Classes / Objetos

- **Composição**
  - Não se baseia no relacionamento entre classes mas sim no relacionamento entre objetos
  - um objeto conhece outro objeto (contém a sua referência numa variável) a quem pode enviar mensagens
- **Forma usual de "pensar em termos de objetos"**

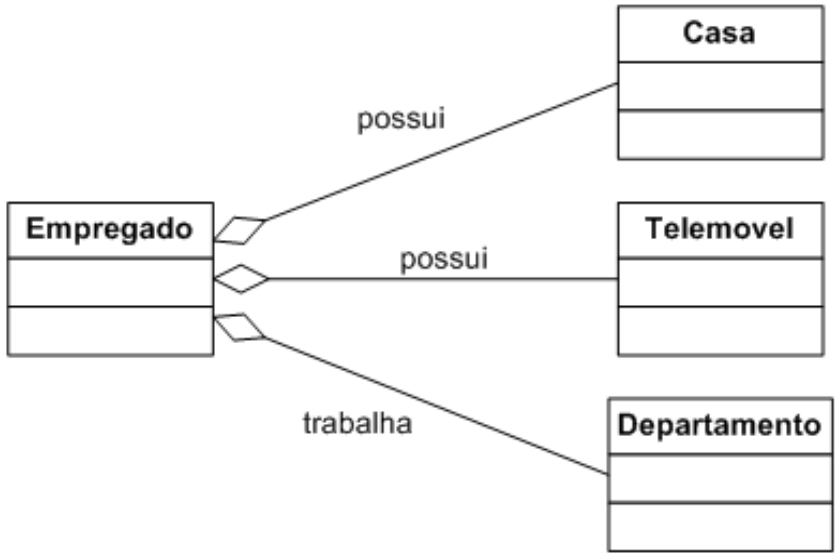


Classes / Objetos

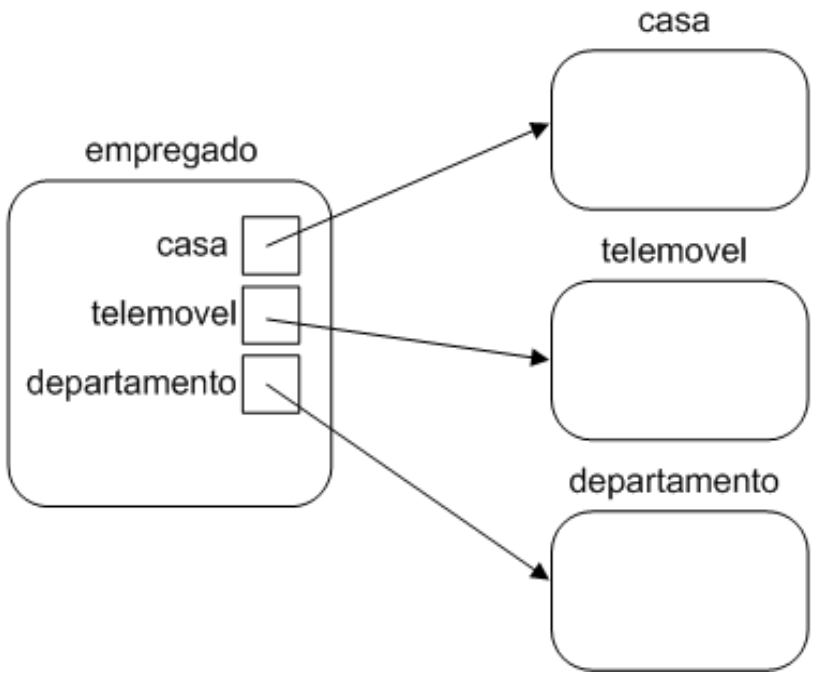
ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- UML vs Notação informal

Relacionamentos entre classes



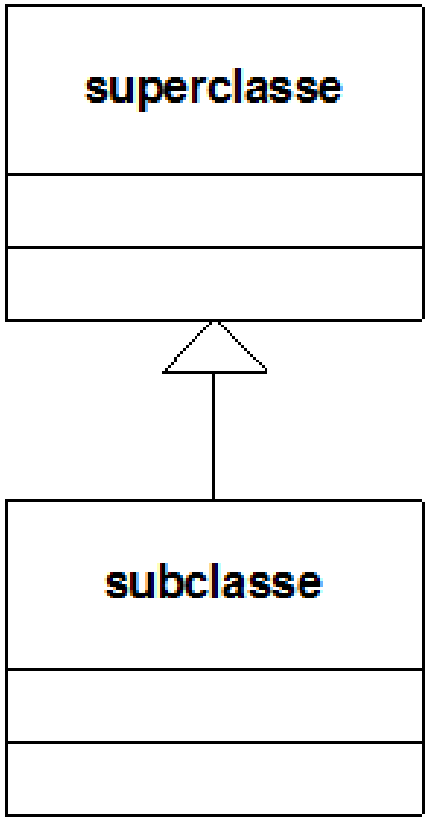
Relacionamentos entre objectos



Classes / Objetos

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- **Herança**
  - As classes relacionam-se entre si de modo a partilharem e enriquecerem as definições das estruturas e comportamentos das suas instâncias
  - Objetivo - reutilização
    - Definir uma vez e reutilizar n vezes
  - Formas de reutilização
    - Herança simples
    - Herança Múltipla

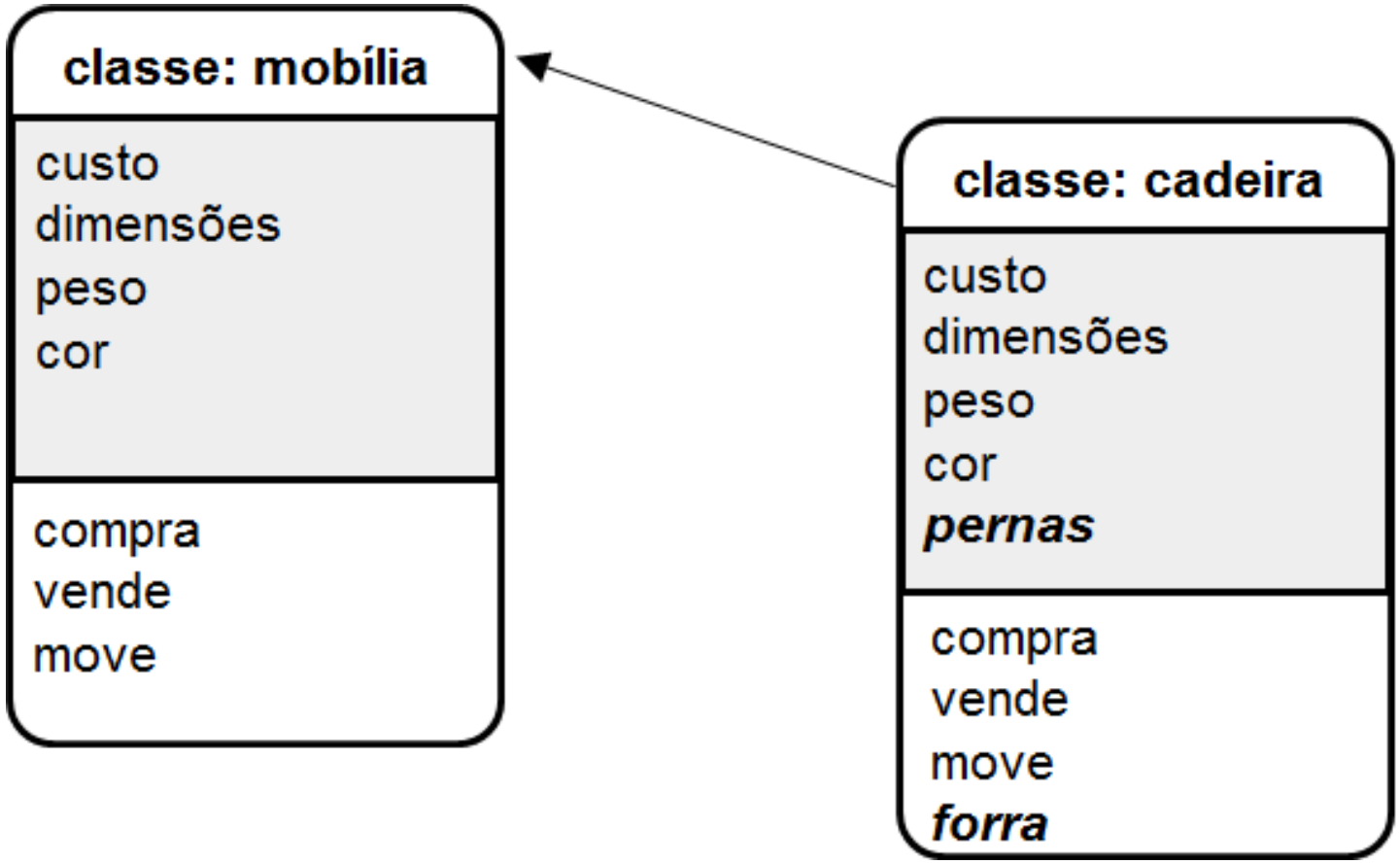


Classes / Objetos

ESCOLA TÉCNICA SUPERIOR PROFISSIONAL IPCA

- **Herança**
  - Uma classe é criada como subclasse de uma superclasse herdando as suas especificações (estado e comportamento) desta última.
  - Uma classe pode acrescentar estado e comportamento ao herdado da sua superclasse bem como alterar o comportamento que herda.

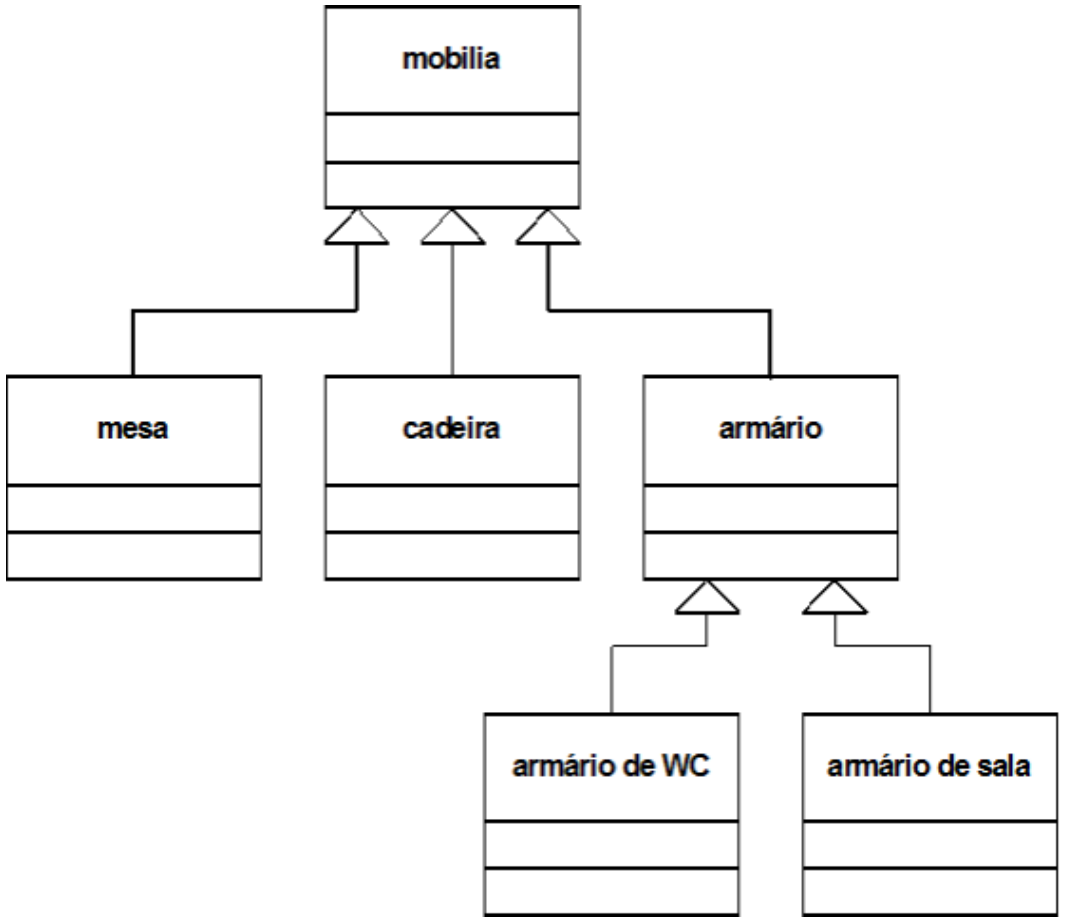
Classes / Objetos



a classe cadeira herda (é subclasse) da classe mobília e pode definir atributos e operações próprias

Classes / Objetos

- **Herança simples**
- uma classe herda a estrutura e comportamento definido na sua superclasse e acrescenta à estrutura e comportamento com as suas próprias definições.
- **Hierarquia de classes**



Polimorfismo

- Conceito de que numa mesma linguagem pode dar origem a comportamentos diferentes.
- Dois objetos diferentes que recebem a mesma mensagem e respondem de formas diferentes.