Fatec Ourinhos

Curso: Análise e Desenvolvimento de Sistemas

Profa. Me. Viviane de Fatima Bartholo

Email: viviane.bartholo@fatecourinhos.edu.br

Fatec
Ourinhos

Orientação a Objetos

Curso: Análise e Desenvolvimento de Sistemas

Professora Viviane de F. Bartholo

Fatec Ourinhos

Na aula de hoje

- Programação Estruturada
- Orientação a Objetos
- Programação Estruturada vs. POO

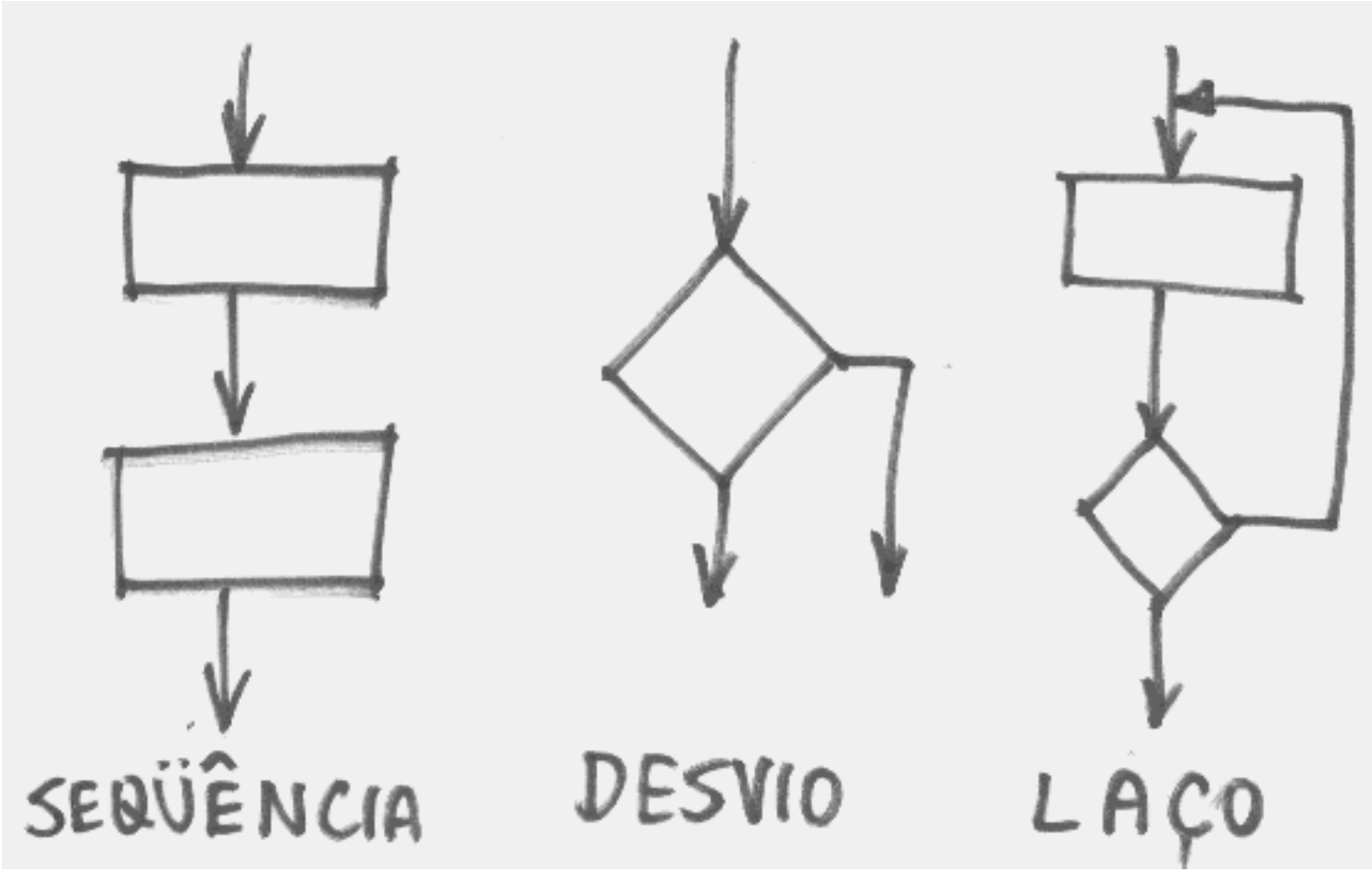
Fatec Ourinhos

- **Paradigma (*pa-ra-dig-ma*)**
  - *Substantivo masculino cujo significado é :*
    1. ***Modelo**, **padrão** ou*
    2. *Termo com o qual Thomas Kuhn (v. kuhniano) designou as realizações científicas (p. ex., a dinâmica de Newton ou a química de Lavoisier) que geram modelos que, por período mais ou menos longo e de modo mais ou menos explícito, orientam o desenvolvimento posterior das pesquisas exclusivamente na busca da solução para os problemas por elas suscitados.*

Fatec Ourinhos

Programação Estruturada

Programação Estruturada



Programação Estruturada

- **A programação estruturada tem como principal foco as ações**
  - Procedimentos e Funções
- **Fornece maior controle sobre o fluxo de execução de um programa**
  - Estruturas de sequência;
  - Estruturas de decisão;
  - Estruturas de repetição.

Fatec Ourinhos

Programação Estruturada

- **As linguagens estruturadas são de entendimento relativamente fácil**
  - Por isso são utilizadas em cursos introdutórios.
- **No entanto, são focadas em como uma tarefa deve ser feita**
  - E não em **o que** deve ser feito.
- **Mistura tratamento de dados e comportamento do programa.**

Fatec Ourinhos

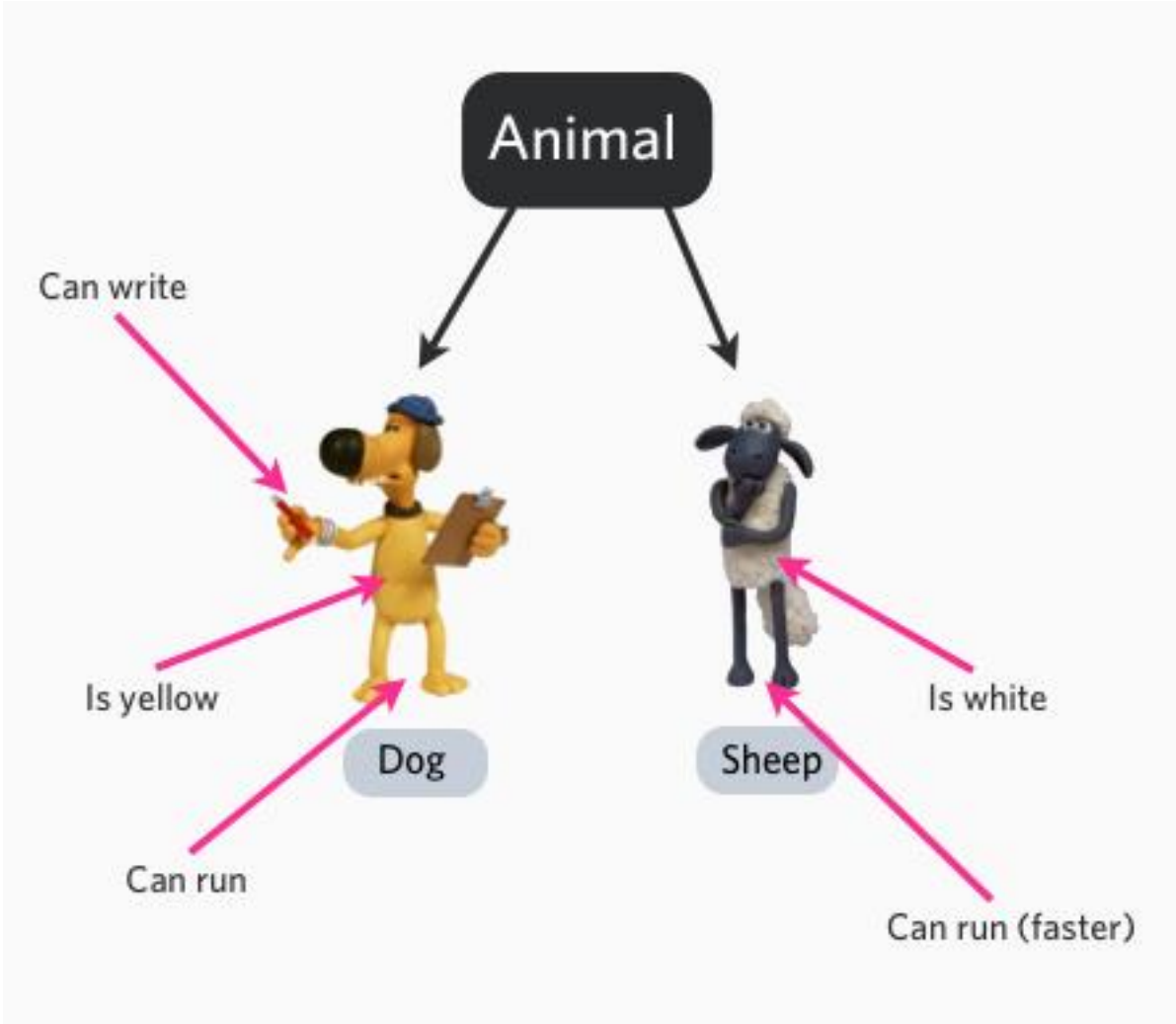
Programação Estruturada

- **A programação estruturada ainda é muito influente**
  - Para cada situação uma ferramenta.
- **Para problemas simples e diretos, ainda é a melhor solução.**

Fatec Ourinhos

Orientação a Objetos

Orientação a Objetos



Fatec Ourinhos

Orientação a Objetos

- **O conceito de Orientação a Objetos data do final da década de 60 e início da década de 70**
  - Simula 67 (60's);
  - Smalltalk (70's);
  - C++ (80's).
- **Surgiu da necessidade de modelar sistemas mais complexos.**

Fatec Ourinhos

Orientação a Objetos

- Como melhor modelar o mundo real utilizando um conjunto de componentes de *software*?
- Considerando que nosso mundo é composto de objetos, porquê não utilizá-los?
- A ideia é modelar utilizando objetos, determinando como eles devem se comportar e como deve interagir entre si.

Fatec Ourinhos

Orientação a Objetos

O paradigma de orientação a objetos visualiza um sistema de software como uma coleção de agentes interconectados chamados ***objetos***.
Cada objeto é responsável por realizar tarefas específicas. É através da interação entre objetos que uma tarefa computacional é realizada.

Um sistema de software orientado a objetos consiste de objetos em colaboração; com o objetivo de realizar as funcionalidades deste sistema. Cada objeto é responsável por tarefas específicas. É através da cooperação entre objetos que a computação do sistema se desenvolve.

Orientação a Objetos

- **Este paradigma de programação tenta ser o mais óbvio, natural e exato possível;**
- **São conceitos essenciais:**
  - Abstração
  - Classes e objetos;
  - Atributos, Métodos e Mensagens;
  - Herança e Associação;
  - Encapsulamento;
  - Polimorfismo;

Fatec Ourinhos

Objetos

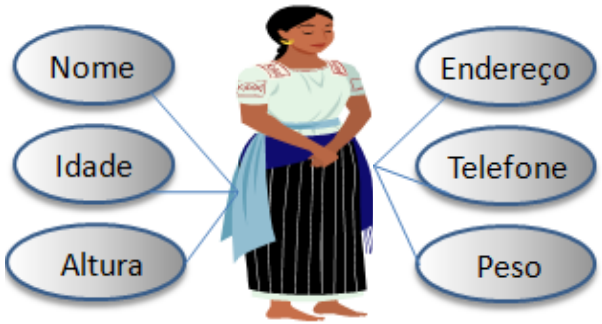
Fatec Ourinhos

Orientação a Objetos

- **abstrair (*abs-tra-ir*)**
  - *Verbo transitivo cujo significado é:*
  - *Separar. V. i. Considerar separadamente. V. p. Concentrar-se. Alhear-se.*
- **Em outras palavras, captar a essência de um problema ou contexto e considerar o que realmente importa.**

Abstração

Identificar todas as informações que atendem a aplicação e ignorar as informações irrelevantes.



Dados abstraidos

Cliente
- nome : texto - endereço : texto - telefone : texto - idade : inteiro

Fatec Ourinhos

Objetos

- **Objetos são a chave para entender a OO;**
- **Se olharmos em nossa volta, encontraremos vários exemplos de objetos reais:**
  - Celular;
  - Mesa;
  - Computador;
  - Janela;
  - Lâmpada;
  - *Etc*.

Fatec Ourinhos

Objetos

- **Os objetos reais possuem duas caracterísicas**
  - Estado (Atributos);
  - Comportamento.
- **Por exemplo, um cachorro**
  - Estado: nome, cor, raça, fome…
  - Comportamento: latindo, abanando o rabo, comendo…
- **Uma bicicleta**
  - Estado: marcha atual, freio, rotação…
  - Comportamento: mudando de marcha, freando…

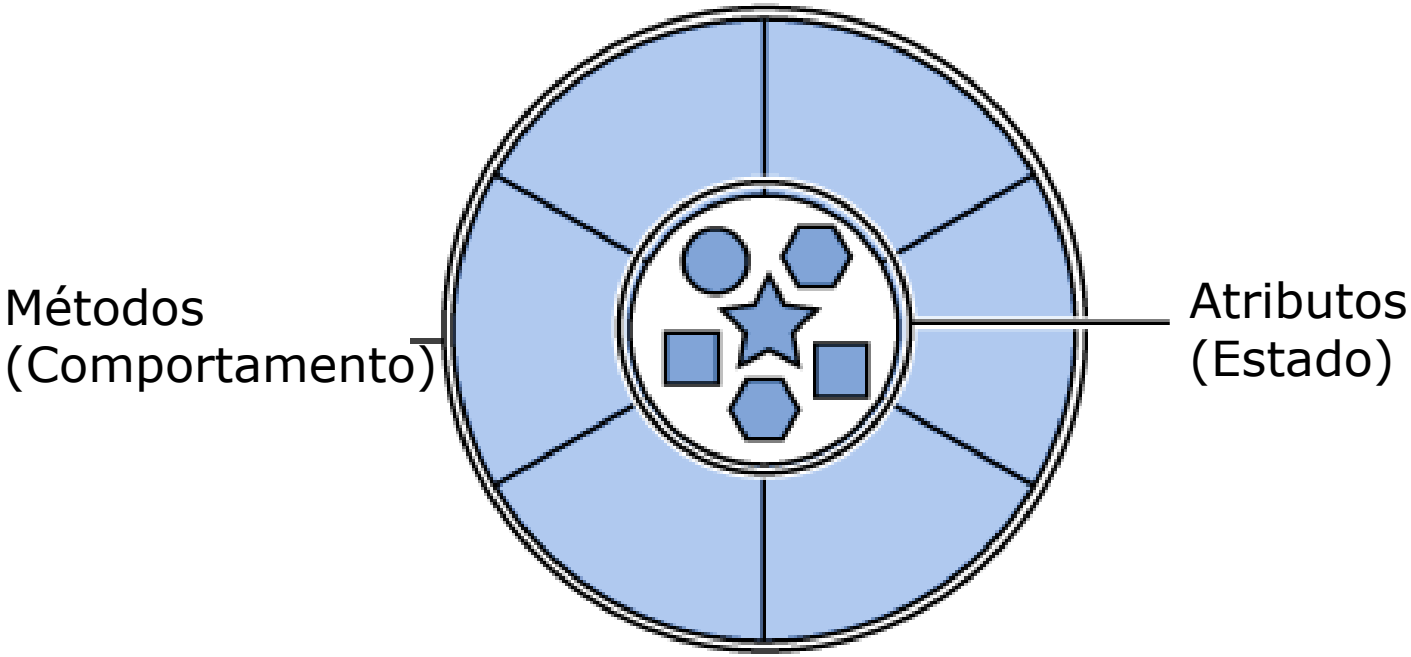
Fatec Ourinhos

Objetos

- **Quais são as características de uma lâmpada?**
- **Quais são as características de um projetor?**
  - E como tratamos a lâmpada do projetor?
- **Objetos variam em complexidade**
  - Porém, os detalhes relevantes dependem do contexto;
  - Esta análise de características é traduzível em orientação a objetos.

Fatec Ourinhos

Objetos

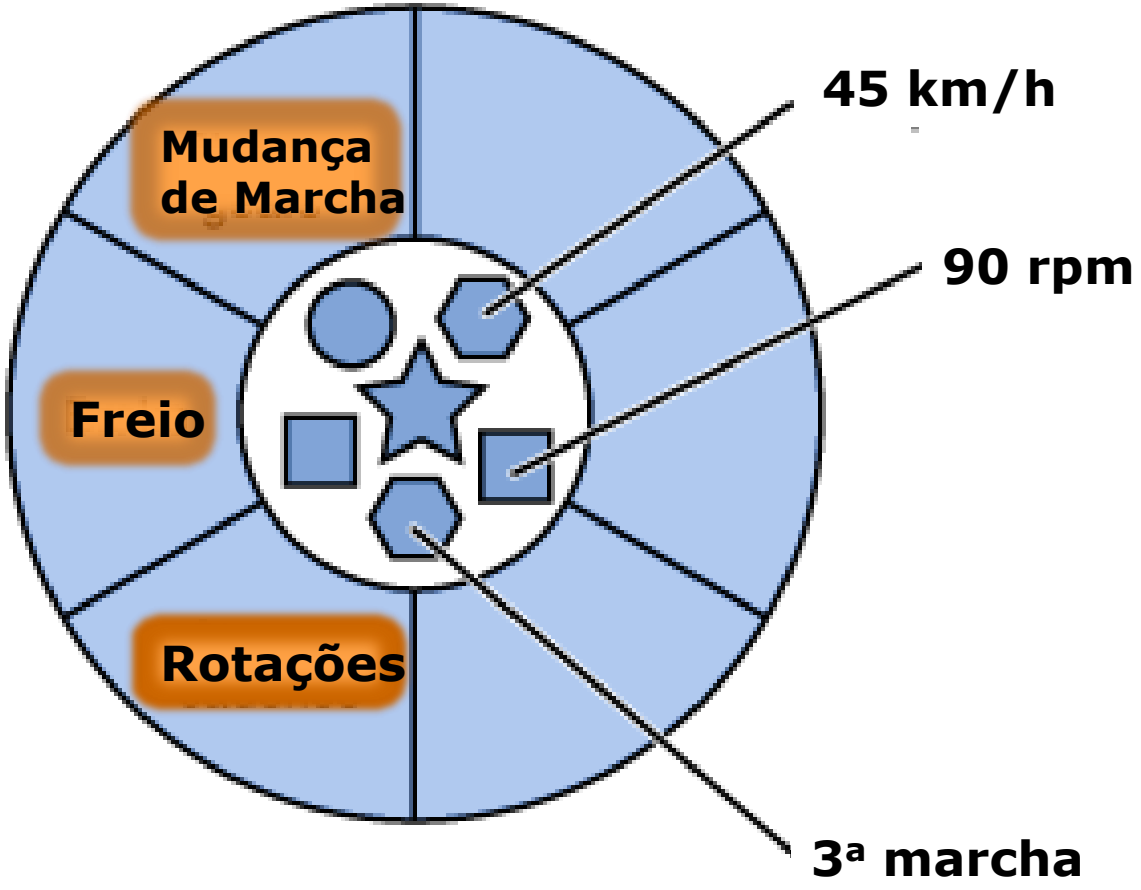


Fatec Ourinhos

Atributos e Métodos

- **Um objeto de *software* é conceitualmente similar aos objetos reais**
- **Objetos armazenam seu estado em atributos**
  - Correspondentes às variáveis em programação estruturada.
- **Objetos expõem seu comportamento através de métodos**
  - Correspondentes às funções em programação estruturada.

Objetos



Fatec Ourinhos

Objetos

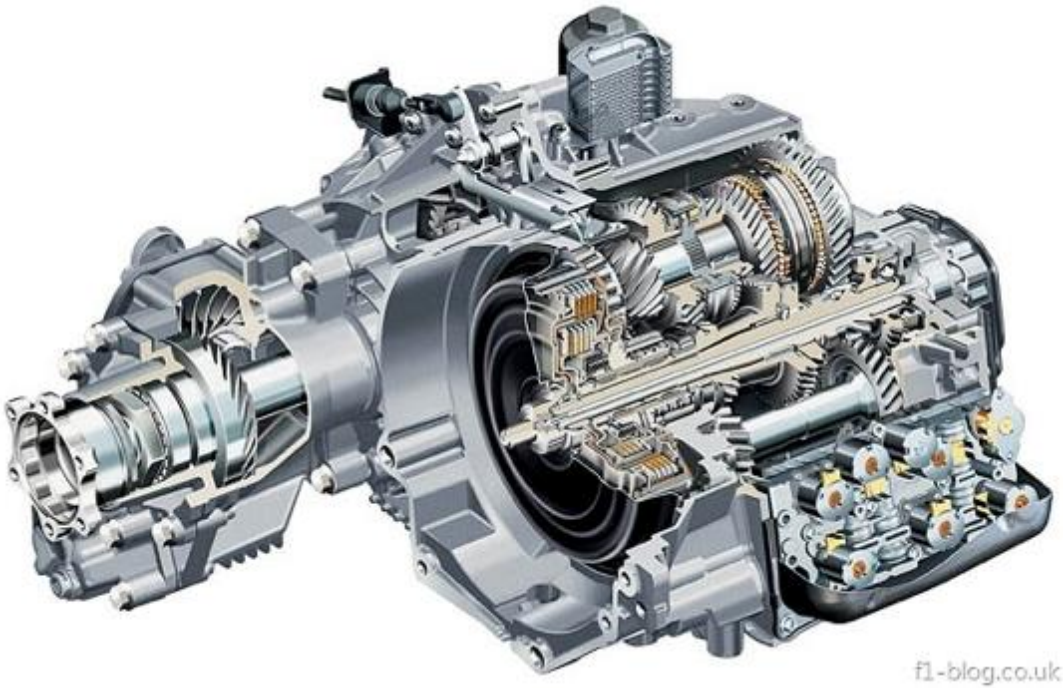
- **Empacotar o código em objetos individuais fornece:**
  - Modularidade
    - Objetos são independentes.
  - Ocultação de informação
    - Os detalhes da implementação de um objeto permanecem ocultos.
  - Reuso
    - Objetos podem ser reutilizados em diferentes programas.
  - Plugabilidade
    - Objetos podem ser substituídos em um programa, como peças.

Fatec Ourinhos

Encapsulamento

Fatec Ourinhos

Encapsulamento de Dados



Fatec Ourinhos

Encapsulamento de Dados

- **Os métodos definem o estado interno de um objeto**
  - E servem como mecanismo primário de comunicação entre objetos.
- **Esconder o estado interno e requerer que toda interação seja feita através de métodos é chamado de encapsulamento de dados**
  - Um princípio fundamental de OO.

Fatec Ourinhos

Encapsulamento de Dados

- **Através do encapsulamento de dados, evitamos alterações acidentais nos atributos de um objeto**
  - Caso haja alguma alteração nos atributos, temos certeza de qual método foi utilizado.
- **A idéia é proteger informações de uma parte da aplicação das demais partes da aplicação**
  - Alterações pontuais podem ser feitas no código sem introdução de *bugs* adicionais em trechos que não tem relação com o trecho alterado.

Fatec Ourinhos

Encapsulamento de Dados

- **Mantendo o estado e provendo métodos para alterar o estado, quem determina como o mundo pode interagir com o objeto é o próprio objeto**
  - O objeto está no controle;
  - Por exemplo, não poderíamos passar a 7ª marcha se o objeto só possuir 6 marchas;
  - Não é o que ocorre no mundo real?

Fatec Ourinhos

Classes

- **No mundo real, encontramos vários objetos de um mesmo tipo**
  - Deve haver centenas de outras bicicletas, do mesmo fabricante e modelo;
  - Cada bicicleta pode ser produzida a partir do mesmo conjunto de projetos e conter as mesmas peças.

Fatec Ourinhos

Classes

- **Em orientação a objetos, dizemos que um objeto é uma instância da classe de objetos conhecida como *Bicicleta*;**
- **Uma classe é o projeto a partir do qual objetos individuais são criados**
  - Ela define os atributos e os métodos correspondentes aos seus objetos.

Fatec Ourinhos

Associação

- **Agregação e Composição são tipos especiais de Associação;**
- **Uma Associação é o mecanismo pelo qual um objeto utiliza os recursos de outro**
  - Pode ser uma associação *simples*
    - "**Usa um**";
    - "Uma Pessoa **usa um** computador".
  - Ou um *acoplamento*
    - "**Parte de**";
    - "O teclado é **parte de** um computador".

Fatec Ourinhos

Herança

- **O relacionamento de Herança define um relacionamento do tipo "é um"**
  - "*Mountain Bike* **é uma** bicicleta".
- **Indica que uma (a *subclasse*) de duas classes relacionadas é uma forma especializada da outra (a *superclasse*)**
  - A superclasse é considerada uma **generalização** da subclasse.

Herança

- **Diferentes tipos de objetos frequentemente possuem semelhanças com outros**
  - Bicicletas *Tandem*;
  - *Mountain bikes*;
  - Bicicletas de corrida.

Fatec Ourinhos

Herança

- **Todas estas bicicletas possuem características de bicicletas**
  - Velocidade atual;
  - Rotação atual;
  - Marcha atual.
- **No entanto, também possuem características diferentes**
  - As *Tandem* possuem dois bancos e guidões;
  - As de corrida possuem guidão angulado;
  - *Moutain bikes* possuem correntes maiores, alterando o peso das marchas.

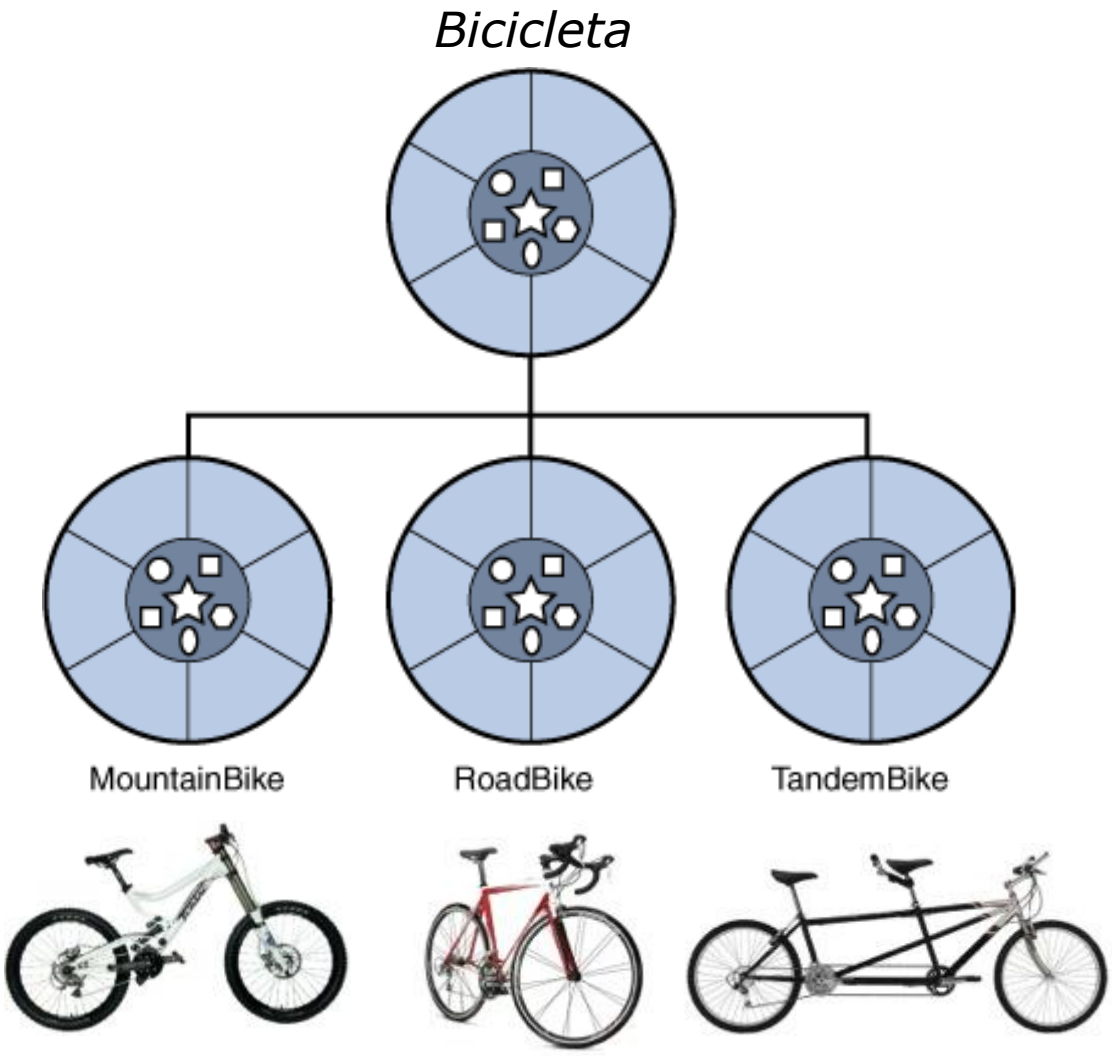
Fatec Ourinhos

Herança

- **A orientação a objetos permite que as classes herdem o estado e comportamento comuns a outras classes**
  - Neste exemplo, a classe *Bicicleta* se torna a **superclasse** de *MountainBike*, *TandemBike* e *RoadBike*
    - Estas agora são consideradas **subclasses**.

Fatec Ourinhos

Herança



Fatec Ourinhos

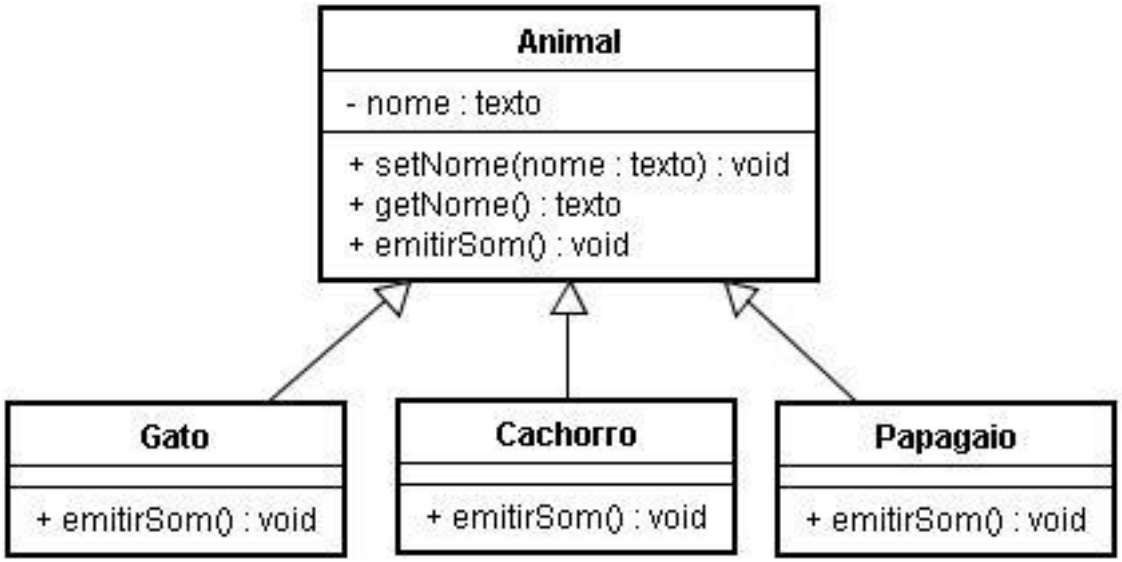
Polimorfismo

- **A definição dos dicionários para polimorfismo se refere a um princípio em biologia no qual um organismo ou espécie pode possuir diferentes formas**
  - Este princípio pode ser aplicado à OO.
- **Podemos utilizar um único nome para definir várias formas distintas**
  - Por exemplo, uma família de funções com o mesmo nome e códigos independentes.

Fatec Ourinhos

Polimorfismo

Capacidade de objetos de diferentes tipos responder a métodos com o mesmo nome, cada um de acordo com seu próprio comportamento.

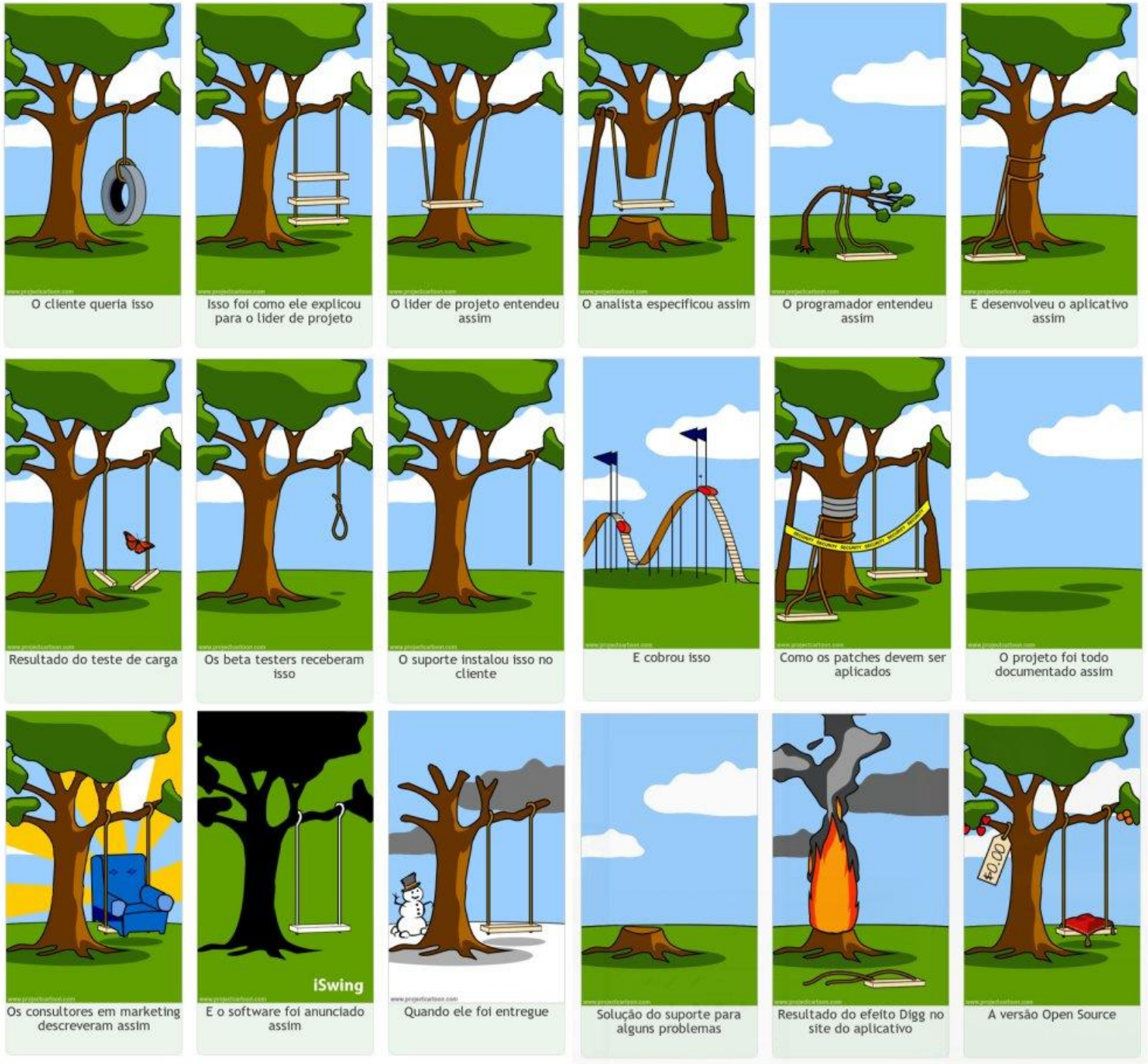


Fatec Ourinhos

Análise e Projeto Orientados a Objetos

Análise e Projeto OO

- **Em breve estaremos programando OO**
  - Como faremos?
  - Ligar o computador e começar a digitar?
    - Funciona para apenas pequenos programas.
  - E se fôssemos contratados para criar um *software* que gerencia os caixas eletrônicos de um grande banco?
  - Ou, se fôssemos trabalhar em uma equipe em que o trabalho é dividido entre 100 pessoas?


O cliente queria isso
Isso foi como ele explicou para o lider de projeto
O lider de projeto entendeu assim
O analista especificou assim
O programador entendeu assim
E desenvolveu o aplicativo assim
Resultado do teste de carga
Os beta testers receberam isso
O suporte instalou isso no cliente
E cobrou isso
Como os patches devem ser aplicados
O projeto foi todo documentado assim
Os consultores em marketing descreveram assim
iSwing
E o software foi anunciado assim
Quando ele foi entregue
Solução do suporte para alguns problemas
Resultado do efeito Digg no site do aplicativo
$0.00
A versão Open Source

Análise e Projeto OO

- **Antes de escrever o código, é necessário analisar os requisitos (*o quê*) de seu projeto e projetar uma solução (*como*) satisfatória**
  - Pode poupar muitas horas de trabalho e dinheiro.
- **Quando esta análise envolve um ponto de vista de OO, chamamos de análise e projeto orientados a objetos.**

Fatec Ourinhos

Análise e Projeto OO

- **Idealmente, membros de um grupo de desenvolvimento devem definir um processo para resolver um determinado problema e uma maneira uniforme para comunicar os resultados deste processo para outro;**
- **Uma linguagem gráfica utilizada para comunicação de qualquer processo de análise e projeto OO é a *Unified Modeling Language*.**

Fatec Ourinhos

Comentários Finais

Fatec Ourinhos

POO

- **A POO nos provê uma melhor organização do código**
  - E também contribui para o reaproveitamento do mesmo.
- **No entanto, o desempenho é geralmente inferior quando comparado com a programação estruturada.**

Fatec Ourinhos

POO

- **A grande dificuldade para compreender a POO é na verdade entender o paradigma de projeto orientado a objetos**
  - A POO se preocupa com os **objetos** e seus **relacionamentos**;
  - A programação estruturada se preocupa com as **ações**.

Fatec Ourinhos

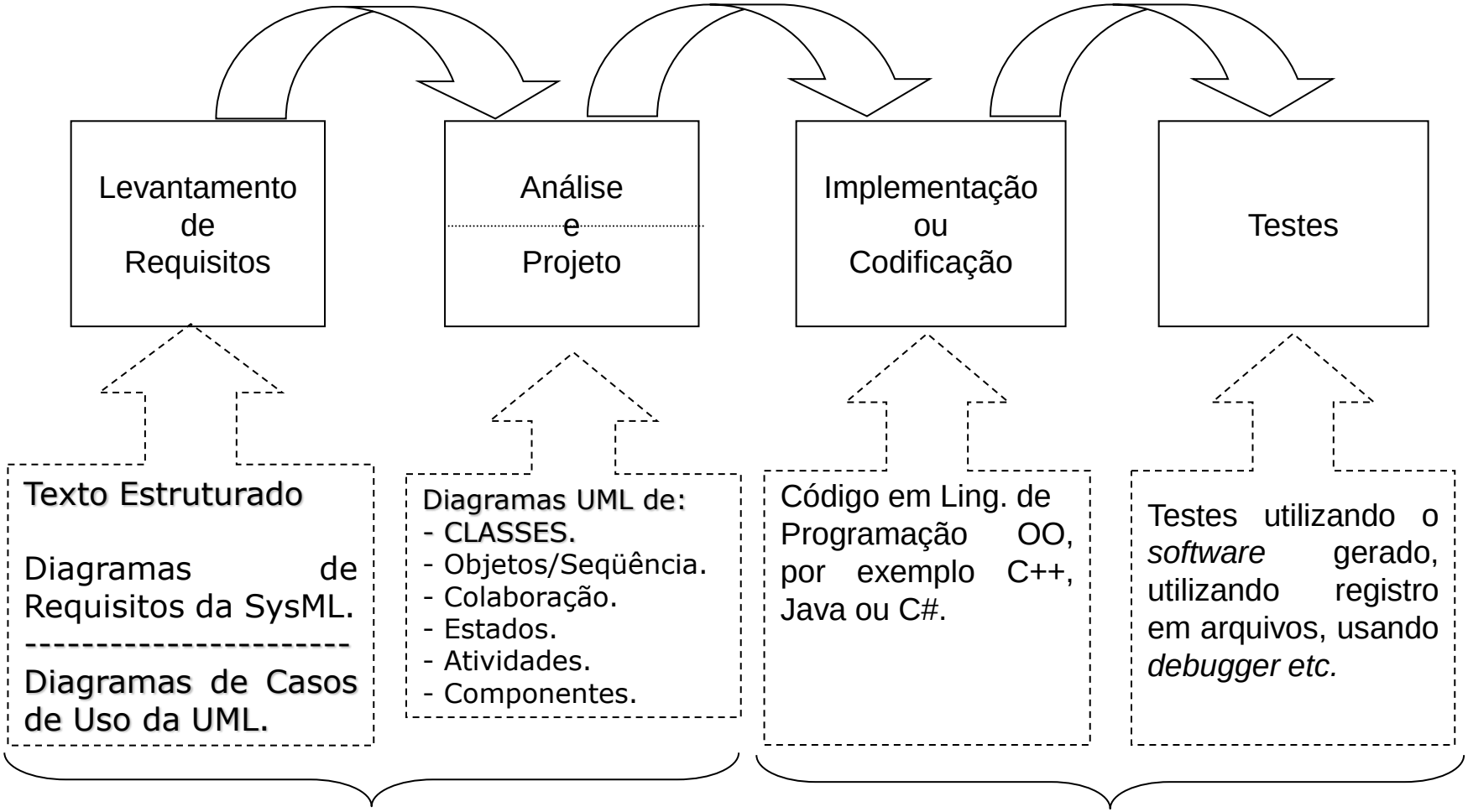
Programação Estruturada vs. POO

1. Funções
2. Variáveis
3. Chamadas de Funções
4. Tipos definidos pelo usuário
5. -
6. -

1. Métodos
2. Objetos/Instâncias
3. Mensagens
4. Classes
5. Herança
6. Polimorfismo

Engenharia de Software OO

Exemplos de Técnicas

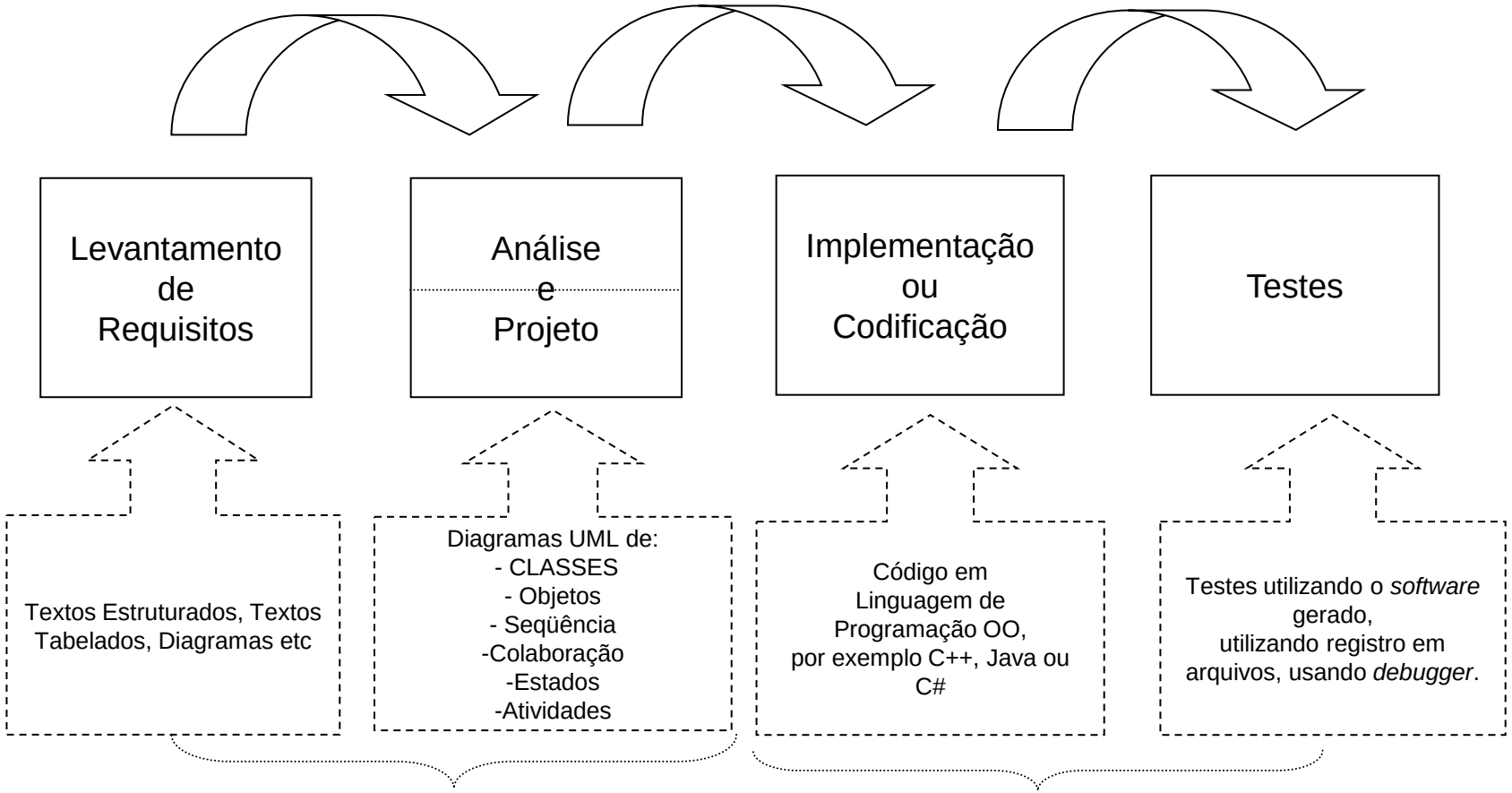


Diagramas da UNIFIED MODELING LANGUAGE (UML)

Linguagem e Ambiente de Programação OO e eventualmente Ambiente de Testes.

Engenharia de Software OO

Exemplos de Ferramentas



Ferramentas CASE: StarUML, **ASTAH**, Rational Rose, System Architect, Together, VisualParadigm, Rhapsody . . .

Ambiente de Programação OO (integráveis as Ferramentas CASE): Microsoft Visual Studio, Microsoft Visual C++ .net Express Edition, Microsoft Visual C++ .net, Microsoft Visual C++, Borland Builder C++, Borland C++, CodeBlocks, Dev C++, G++...

DÚVIDAS??

FIM

Fatec Ourinhos

Referência Bibliográfica

- Guedes, G., UML 2.0 - Uma Abordagem Prática . Rio de Janeiro : Novatec /2009.
- Booch, G., Rumbaugh, J. and Jacobson, I., *Unified Modeling Language User Guide*, 2nd Edition, Addison-Wesley Object Technology Series, 2005.
- P. J. DEITEL, H. M. DEITEL. C++ Como Programar. Quinta edição. Pearson, 2006.
- P. J. DEITEL, H. M. DEITEL. Java Como Programar. Oitava edição. Pearson, 2010.
- B. MEYER. Object-Oriented Software Construction. Segunda Edição. PrentticeHall, 1997.
- Nota de aula de Marco Antonio Moreira de Carvalho. Disponivel em http://www.decom.ufop.br/marco/ensino/bcc221/material-das-aulas