


PROF. EDUARDO ENARI

Sobre programação orientada a objetos


Paradigmas da programação




Programação estruturada

- A partir de funções que o sistema deveria realizar.
 - Requisitos funcionais => definindo o sistema a partir de funções que serão implementadas na forma de procedimentos.
 - Dados eram um complemento das funções / procedimentos do sistema.
- 
- A teal-colored decorative curve is located in the bottom right corner of the slide, starting from the bottom edge and curving upwards and to the left.

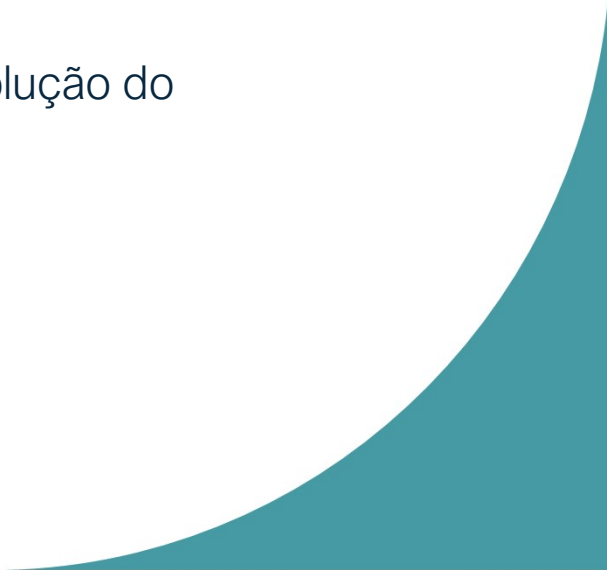
Exemplo de modelagem para programação estruturada

- Sistema para gestão de frotas de veículos
 - Cadastro de veículos (incluir, alterar e dar baixa no veículo)
 - Cadastro de motoristas
 - Cadastro de oficinas
 - Cadastro de postos de combustível
 - Levantamento de dados
 - Dados dos veículos
 - Dados dos motoristas
- 
- A teal-colored decorative curve is located in the bottom right corner of the slide, starting from the bottom edge and curving upwards and to the left.

Exemplo

- Procedimento Cadastro de Veículos
 - Inclusão
 - Busca
 - Alteração
 - Baixa
- 
- A teal-colored decorative curve starts from the bottom right corner and extends upwards and to the left, curving towards the center of the slide.

Mais um Exemplo

- Problema: Resolver uma equação de 2º Grau.
 - Equação de Báskara.
 - a, b, c (entradas)
 - X1 e/ou X2 (soluções)
 - Em situações como esta, o algoritmo já bem conhecido, comanda a solução do problema e a forma como se implementa o programa.
- 
- A teal-colored decorative curve starts from the bottom right corner and extends upwards and to the left, ending near the center of the right edge of the slide.


Programação orientada a objetos

- Organizamos as coisas a partir de dois aspectos:
 - Classificação
 - O que define os integrantes de uma classe? Características (lista de atributos)
 - Atributos são os valores que identificam um indivíduo da classe em relação aos demais indivíduos da classe.
 - Indivíduos da classe são chamados objetos da classe.
 - Associação
 - A partir das características do objeto, identifica-se a que classe ele pertence.


Exemplo de modelagem orientada a objetos => TAD

- Sistema de gestão de frota de veículos:
 - Classe veículos => haverão diversos veículos a serem geridos pelo sistema
 - Características: modelo, fábrica, ano, etc.
 - Atributos: modelo: saveiro, fábrica: volkswagen, ano: 2016, Renavan: XXXX, Placa: XXXX
 - Classe motoristas => diversos motoristas a serem geridos pelo sistema
 - Classe oficinas
 - Classe postos
- Motorista X dirige veículo Y => deseja-se manter registro dessa ocorrência
 - Classe registroUsoVeículoPorMotorista.
 - Características: Data, Motorista, Veículo, quilometragem

Qual a vantagem de usar POO?

- “Diminuir o GAP entre a programação e o problema”
 - Classes que forem modeladas poderão ser programadas
 - TAD (classes do mundo real) => TD (Classes do programa)
 - Classes são novos TD para criação de objetos no seu programa
- 

Histórico da POO

- POO começou com uma linguagem de programação chamada SIMULA, na década de 1960!
 - Possibilitava o uso de objetos e se prestava para desenvolver simulações de processos.
 - Smaltalk (usei em 1992)
 - C++ (C com classes)
 - Object Pascal (Pascal com Classes)
 - Java
 - Ruby
 -
- 
- A teal-colored decorative curve starts from the bottom right corner and extends upwards and to the left, ending near the middle of the right edge of the slide.