

An abstract, light blue geometric pattern consisting of interconnected lines and star-like shapes, resembling a complex network or a stylized molecular structure, is positioned in the upper left corner of the slide.

Programação Orientada para Objectos

Programação estruturada

- Forma
 - Instâncias de tipos escalares ou matriciais
 - Instruções organizadas em estruturas de controlo
- Sem modularização
- Melhorias
 - Melhora estruturação do fluxo de controlo dos programas
 - Possibilita raciocínio formal acerca de programas
 - Aumenta legibilidade dos programas face ao passado (programação “esparguete”)

Programação procedimental

- Forma
 - Instâncias de tipos escalares ou vectoriais
 - Instruções organizadas em estruturas de controlo
 - Instruções e estruturas de controlo organizadas em **rotinas**
- Modularização em rotinas
 - **Funções** – Devolvem resultado de um cálculo
 - **Procedimentos** – Agem sobre dados, modificando-os
- Melhorias
 - Possibilita encapsulamento
 - Facilita reutilização
 - Melhora localização de erros e facilita manutenção
 - Possibilita maximização da coesão e minimização das ligações

Programação baseada em objectos ou centrada nos dados

- Forma
 - Instâncias de tipos escalares ou vectoriais, ou de TAD (tipos abstractos de dados)
 - Estruturas de controlo e rotinas
 - Rotinas relacionadas organizadas em TAD
- Modularização
 - TAD – Conjunto de dados e operações
 - Operações – Rotinas que operam sobre instâncias de TAD
- Melhorias
 - Melhora encapsulamento (dados e operações relacionadas)
 - Possibilita ocultação dos dados
 - Muda perspectiva sobre os problemas

Programação orientada por objectos

- Forma
 - Instâncias de **classes** (**objectos**), de tipos escalares ou vectoriais, ou de TAD (tipos abstractos de dados)
 - Estruturas de controlo e rotinas
 - Rotinas relacionadas organizadas em TAD
 - Operações relacionadas organizadas em classes
- Modularização
 - Classes – Modelos para objectos com um dado comportamento
 - TAD – Conjunto de dados e operações
 - Operações – Rotinas que operam sobre instâncias de TAD
 - **Métodos** – Implementação das operações
- Melhorias
 - Introduz noções de extensão e especialização
 - Muda drasticamente perspectiva sobre os problemas

Programação orientada por objectos: classes

- Paradigmas usados
 - Programação estruturada – Controlo de fluxo
 - Programação procedimental – Rotinas
 - Programação centrada nos dados – TAD e operações
- Vantagens
 - Melhor modularização
 - Melhor encapsulamento
 - Maior possibilidade de reutilização
 - Possibilidade de extensão e especialização
 - Maior expressividade
 - Maior flexibilidade
 - Maior robustez

Programação orientada por objectos: encapsulamento

- Interface:
 - Operações – Implementadas em um ou mais métodos
 - **Propriedades** – Podem ou não ser implementadas usando atributos
- Implementação
 - Métodos – Implementação de operações
 - **Atributos** – Dados que fazem parte da implementação da classe

Programação orientada por objectos: abordagem

- Tudo (ou quase) são objetos
- Objectos têm **responsabilidades, comportamentos** e propriedades
- Organização de programas reflecte realidade...

Programação orientada por objectos: análise e desenho

- Análise do problema
 - Que objectos existem?
 - Que **responsabilidades** têm?
 - Como **colaboram**?
 - Como classificar os objetos?
- Desenho da solução
 - Que classes **definir**?
 - Que objectos **construir**?
 - Que responsabilidades lhes atribuir?
 - De que forma os fazer colaborar?

Referências

- Y. Daniel Liang, "Introduction to Java Programming" 7th Ed. Prentice-Hall, 2010

Sumário

- Programação Orientada pra Objetos