

---

**Departamento de Engenharia Elétrica  
Processo Seletivo para Professor Substituto**

**Área de Conhecimento: Engenharia de Software e Programação Orientada a Objetos**

**PROGRAMA**

**A) Engenharia de Software**

- I – Introdução à Engenharia de Software
  - I.1 – Motivação
  - I.2 – Definição
  - I.3 – Histórico
  - I.4 – Conceitos fundamentais
  - I.5 – Mitos relativos ao software
  - I.6 – Responsabilidade profissional e ética do Engenheiro de Software
- II - Processos de Software
  - II.1 – Ciclo de vida de software
  - II.2 – Modelos de processo de software
  - II.3 – Métodos ágeis de desenvolvimento de software
    - II.3.1 – Manifesto ágil
    - II.3.2 – Programação extrema (XP)
    - II.3.3 – Scrum
    - II.3.4 – DevOps
    - II.3.5 – Gestão ágil de projetos de software
- III – Engenharia de Requisitos
  - III.1 – Requisitos
  - III.2 – Classificação para requisitos
  - III.3 – O processo de engenharia de requisitos
    - III.3.1 – Desenvolvimento de requisitos
    - III.3.2 – Métodos para especificação de requisitos
    - III.3.3 – Gerência de requisitos
- IV – Análise, Projeto e Construção de Software
  - IV.1 – Análise orientada a objetos
  - IV.2 – Arquitetura de software
  - IV.3 – Modelagem de software orientada a objetos
  - IV.4 – Projeto orientado a objetos
  - IV.5 – Construção de software orientado a objetos
- V – Qualidade de software
  - V.1 – Gerenciamento da qualidade de software
  - V.2 – Verificação e validação de software
    - V.2.1 – Revisões e inspeções
    - V.2.2 – Testes de software
  - V.3 – Medições e métricas de software
  - V.4 – Introdução aos métodos formais de qualidade de software
- VI - Organização de equipes de desenvolvimento e gerência de pessoal

**B) Programação Orientada a Objetos**

---

Unidade I – Gerenciando a Complexidade

- I.1. A complexidade inerente dos sistemas de software
- I.2. A estrutura de sistemas complexos
- I.3. Trazendo ordem ao caos
  - I.3.1. Decomposição
  - I.3.2. Hierarquia
  - I.3.3. Abstração

Unidade II - A Modelagem OO

- II.1. A evolução dos modelos orientados a objetos
- II.2. A abstração de objetos: encapsulamento, interface e implementação
- II.3. Reutilizando a implementação
- II.4. Reutilizando a interface por meio de herança
- II.5. Objetos intercambiáveis: polimorfismo
- II.6. Introdução à Análise e ao Projeto Orientados a Objetos

Unidade III - Classes e Objetos

- III.1. Paradigmas de programação
- III.2. Implementando classes e objetos em C++
- III.3. Atributos e métodos: controle de acesso e encapsulamento
- III.4. Inicialização e destruição
- III.5. Sobrecarga de funções e argumentos default
- III.6. Constantes e controle de visibilidade
- III.7. Ponteiros, referências, atributos dinâmicos, gerenciamento de memória e o construtor de cópia
- III.8. Sobrecarga de operadores e conversão de tipos

Unidade IV - Herança e Composição

- IV.1. Composição
- IV.2. Herança simples
- IV.3. Combinando composição e herança
- IV.4. Herança múltipla
- IV.5. Polimorfismo e funções virtuais
- IV.6. Classes abstratas

Unidade V - Outros Mecanismos da Linguagem C++

- V.1. Classes parametrizadas (templates)
- V.2. Exceções

## **Bibliografia - Engenharia de Software**

- Ian Sommerville. Engenharia de Software. 10ª edição. Pearson Universidades, 2019.
- Roger S. Pressman e Bruce R. Maxim. Engenharia de Software: uma abordagem profissional. 9ª edição. AMGH, 2021.
- Valente, Marco Túlio. Engenharia de Software Moderna. 2020. Disponível em <https://engsoftmoderna.info>.

## **Bibliografia- Programação Orientada a Objetos**

- Bruce Eckel, Thinking in C++ vols 1 & 2, 2nd Edition, Prentice Hall, 2000.
- Harvey M. Deitel & Paul J. Deitel, C++ como programar, 5ª edição. Pearson Universidades, 2006.

- 
- Grady Booch, James Rumbaugh e Ivar Jacobson. UML: Guia do Usuário. 2ª edição. Elsevier Academic, 2017.
  - Bjarne Stroustrup. A Tour of C++, 2nd Edition, Addison-Wesley Professional, 2018.
  - Terry Quatrani, Jim Palistrant. Visual Modeling with IBM Rational Software Architect and UML. IBM Press, 2006.
  - James Rumbaugh, Ivar Jacobson e Grady Booch. The Unified Modeling Language Reference Manual, 2nd Edition. Addison-Wesley Professional, 2004.
  - Grady Booch. Object-Oriented Analysis and Design with Applications 3rd. ed. Addison-Wesley Professional, 2007.
  - Antonio Mendes da Silva Filho. Introdução à Programação Orientada a Objetos com C++, Elsevier, 2010.