

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Engenharia da Computação – 3ª série

***Apresentação*
*(L1/1 – L2/1)***

2024

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Horário

Terça-feira: 2 x 2 aulas/semana

- L1/1 (07h40min-09h20min): *Prof. Calvetti*;
- L1/2 (09h30min-11h10min): *Prof. Calvetti*;
- L2/1 (07h40min-09h20min): *Prof. Igor Silveira*;
- L2/2 (11h20min-13h00min): *Prof. Calvetti*.

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Ementa

- Conceitos fundamentais de Engenharia de *Software* e do paradigma da Orientação a Objetos – OO;
- Ambientes de desenvolvimento (jdk/Eclipse/NetBeans/JGrasp);
- Estrutura de programas com Java;
- Tipos de dados primitivos e classes fundamentais em Java;
- Expressões e operadores em Java;
- Estruturas de programação em Java;
- Variáveis indexadas em Java;
- Tipos e *Wrappers* em Java;

Continua...

Ementa (continuação)

- Programação Orientada a Objetos – POO em Java:
Classes, métodos e atributos, encapsulamento e ocultação da informação, construtores, inicialização e destruição de objetos, uso de herança/polimorfismo, interfaces e classes abstratas;
- Tratamento de exceções Java;
- Criando aplicações gráficas com Java;
- Programação com Bancos de Dados – BD em Java (JDBC);
- Programação com rede;
- Programação concorrente;

Continua...

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Ementa (continuação)

- Coleções em Java e manipulação de arquivos;
- Introdução a desenvolvimento web, HTML5, CSS3, *JavaScript*, *Bootstrap*, *jQuery* e outros *Frameworks*;
- Arquitetura Cliente-Servidor (*cliente-server*) para uma aplicação *WEB*; e
- Introdução a *Back-End*.

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Objetivos

Conhecimentos:

1. Entender o paradigma da POO;
2. Familiarizar-se com linguagens de programação OO, estudando suas sintaxes e semânticas;
3. Aprender a utilizar *frameworks* e bibliotecas OO para resolver problemas comuns de computação, tais como interfaces gráficas, suporte a BD, comunicação em rede e programação concorrente;
4. Aprender a estruturar a solução de problemas com projeto e implementação OOs;

Continua...

Objetivos

Conhecimentos (continuação):

5. Conhecer o significado e a necessidade de tratar o *software* como um Produto de Engenharia, utilizando métodos para seu desenvolvimento;
6. Adquirir familiaridade com processos de desenvolvimento de sistemas em geral;
7. Entender o fluxo básico de desenvolvimento de aplicações na *WEB*;
8. Desenvolver portais (*sites*) com *frameworks* atuais;
9. Compreender a arquitetura Cliente-Servidor para Sistemas *WEB*.

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Objetivos

Habilidades:

1. Trabalhar em equipe;
2. Programar sistemas OOs com linguagens relevantes;
3. Adquirir uma visão geral do processo de desenvolvimento de Sistemas OOs;
4. Implementar sistemas de *software* OOs;
5. Entender como programar OO com linguagens de programação de significância no mercado, como o Java.

Objetivos

Atitudes:

1. Ter iniciativa para solução de problemas;
2. Ter iniciativa de pesquisar soluções de problemas existentes em um projeto de *software*;
3. Adquirir uma postura de trabalho em grupo;
4. Adquirir autonomia na tomada de decisões e execução do trabalho.

Instrumentos de Avaliação

- Provas;
- Relatórios;
- Trabalhos individuais e em grupos;
- Seminários.

Avaliação

- Disciplina anual, com 2 (dois) trabalhos (um por semestre) e 4 (quatro) provas (duas por semestre);
- $MT = (T_1 + T_2) / 2$
- $MP = (2P_1 + 2P_2 + 3P_3 + 3P_4) / 10$
- Peso da MP (k_p): 0,5 e Peso da MT (k_t): 0,5
- $MF = 0,5 MP + 0,5 MT$

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Bibliografia Básica

- MILETTO, Evandro M.; BERTAGNOLLI, Silvia de Castro. Desenvolvimento de software II: introdução ao desenvolvimento web com HTML, CSS, javascript e PHP (Tekne). Porto Alegre: Bookman, 2014. E-book. Referência Minha Biblioteca: <https://integrada.minhabiblioteca.com.br/#/books/9788582601969>
- WINDER, Russel; GRAHAM, Roberts. Desenvolvendo Software em Java, 3ª edição. Rio de Janeiro: LTC, 2009. E-book. Referência Minha Biblioteca: <https://integrada.minhabiblioteca.com.br/#/books/978-85-216-1994-9>
- DEITEL, Paul; DEITEL, Harvey. Java: how to program early objects. Hoboken, N. J: Pearson, c2018. 1234 p. ISBN 9780134743356.

Continua...

Bibliografia Básica (continuação)

- HORSTMANN, Cay S; CORNELL, Gary. Core Java. SCHAFRANSKI, Carlos (Trad.), FURMANKIEWICZ, Edson (Trad.). 8. ed. São Paulo: Pearson, 2010. v. 1. 383 p. ISBN 9788576053576.
- LIANG, Y. Daniel. Introduction to Java: programming and data structures comprehensive version. 11. ed. New York: Pearson, c2015. 1210 p. ISBN 9780134670942.
- TURINI, Rodrigo. Desbravando Java e orientação a objetos: um guia para o iniciante da linguagem. São Paulo: Casa do Código, [2017]. 222 p. (Caelum).

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Bibliografia Complementar

- HORSTMANN, Cay. Conceitos de Computação com Java. Porto Alegre: Bookman, 2009. E-book. Referência Minha Biblioteca: <https://integrada.minhabiblioteca.com.br/#/books/9788577804078>
- MACHADO, Rodrigo P.; FRANCO, Márcia H. I.; BERTAGNOLLI, Silvia de Castro. Desenvolvimento de software III: programação de sistemas web orientada a objetos em java (Tekne). Porto Alegre: Bookman, 2016. E-book. Referência Minha Biblioteca: <https://integrada.minhabiblioteca.com.br/#/books/9788582603710>
- BARRY, Paul. Use a cabeça! Python. Rio de Janeiro: Alta Books, 2012. 458 p. ISBN 9788576087434.

Continua...

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Bibliografia Complementar (continuação)

- LECHETA, Ricardo R. Web Services RESTful: aprenda a criar Web Services RESTfulem Java na nuvem do Google. São Paulo: Novatec, c2015. 431 p. ISBN 9788575224540.
- SILVA, Maurício Samy. JQuery: a biblioteca do programador. 3. ed. rev. e ampl. São Paulo: Novatec, 2014. 544 p. ISBN 9788575223871.
- SUMMERFIELD, Mark. Programação em Python 3: uma introdução completa à linguagem Python. Rio de Janeiro: Alta Books, 2012. 506 p. ISBN 9788576083849.

Continua...

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Bibliografia Complementar (continuação)

- YING, Bai. Practical database programming with Java. New Jersey: John Wiley & Sons, c2011. 918 p.
- ZAKAS, Nicholas C. The principles of object-oriented JavaScript. San Francisco, CA: No Starch Press, c2014. 97 p. ISBN 9781593275402.

ECM251 – Linguagens de Programação I

Aula 01 – L1/1 e L2/1

Softwares Utilizáveis

- Java JDK (versão 11 ou posterior) - Instalador:
<https://docs.aws.amazon.com/corretto/latest/corretto-11-ug/downloads-list.html>
- Google Chrome - Instalador: <https://www.google.com/chrome/>
- JGrasp (jGRASP 2.0.6_10 or higher (August 10, 2022) - requires Java 8 or higher)
https://spider.eng.auburn.edu/user-cgi/grasp/grasp.pl?dl=download_jgrasp.html
- Visual Studio Code - Instalador: <https://code.visualstudio.com/>
- IntelliJ Community Edition - Instalador:
<https://www.jetbrains.com/idea/download/download-thanks.html?platform=windows&code=IICO>

Continua...

Softwares Utilizáveis (continuação)

- Eclipse IDE (Java SE Edition ou superior) - Instalador:
<https://www.eclipse.org/downloads/packages/release/2018-12/r/eclipse-ide-java-developers>
- Netbeans - Instalador:
<http://ftp.unicamp.br/pub/apache/incubator/netbeans/incubating-netbeans/incubating-10.0/incubating-netbeans-10.0-bin.zip>
- MySQL Workbench - Instalador:
<https://dev.mysql.com/downloads/workbench/>
- Apache Tomcat (Tomcat 8.5.85 Released or higher)
<https://tomcat.apache.org/>

Apresentação do Professor

- Pessoal;
- Formação Acadêmica;
- Experiência Profissional; e
- Expectativa.

Apresentação dos Alunos

- Pessoal;
- Formação Acadêmica;
- Experiência Acadêmica;
- Experiência Profissional, se existir;
- Expectativa Acadêmica; e
- Expectativa Profissional.

ECM251 – Linguagens de Programação I

Aula 01

PAUSA...

ECM251 – Linguagens de Programação I

Aula 01 – L1/2 e L2/2

Engenharia da Computação – 3ª série

***Apresentação*
*(L1/2 – L2/2)***

2024

Conceitos

- Engenharia de Software;
- Paradigma da Orientação a Objeto – OO;
- Programação Orientada a Objeto – POO.

Método Socrático

- O que é o Método Socrático?

*“Só sei que nada sei...”
- Sócrates*



Método Socrático

- O que é o Método Socrático?
- Ele tem esse nome graças ao seu criador, Sócrates, famoso filósofo grego, que ensinava fazendo perguntas após perguntas;
- Sócrates procurava expor contradições nos pensamentos e ideias dos seus estudantes, para então guiá-los a conclusões sólidas e sustentáveis.

ECM251 – Linguagens de Programação I

Aula 01 – L1/2 e L2/2

Pergunta

- O que é *Software*?



Pergunta

- O que é *Software*?
- Ou Programa, é o conjunto de componentes lógicos de um computador ou sistema de processamento de dados;
- Rotina ou conjunto de instruções que controlam o funcionamento de um computador;
- Esses artefatos incluem programas executáveis em computador de qualquer porte ou arquitetura.

ECM251 – Linguagens de Programação I

Aula 01 – L1/2 e L2/2

Pergunta

- Quem desenvolve *Software*?

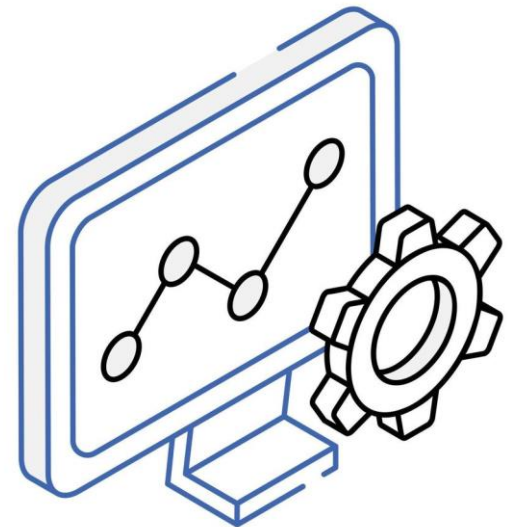


Pergunta

- Quem desenvolve *Software*?
- Os engenheiros de *software* o criam e dão suporte;
- Os engenheiros de *software* aplicam o processo de engenharia de *software*.
- Os programadores o programam e testam o programado;
- Os testadores o testam de diferentes formas.

Pergunta

- O que é Engenharia de *Software*?



Pergunta

- O que é Engenharia de *Software*?
- Área da engenharia e da computação voltada à especificação, desenvolvimento, manutenção e criação de software, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando organização, produtividade e qualidade;
- Processo formado por um conjunto de métodos, ou práticas, e uma série de ferramentas que possibilitam aos profissionais desenvolverem *software* de altíssima qualidade.

ECM251 – Linguagens de Programação I

Aula 01 – L1/2 e L2/2

Pergunta

- Por que a Engenharia de *Software* é importante?



Pergunta

- Por que a Engenharia de *Software* é importante?
- Capacita os profissionais para o desenvolvimento de sistemas complexos dentro do prazo e com alta qualidade;
- Impõe disciplina a um trabalho que pode se tornar caótico;
- Permite que se produzam *softwares* de computador adaptados às suas abordagem, da maneira mais conveniente às suas necessidades.

Pergunta

- Quais são as etapas da Engenharia de *Software*?

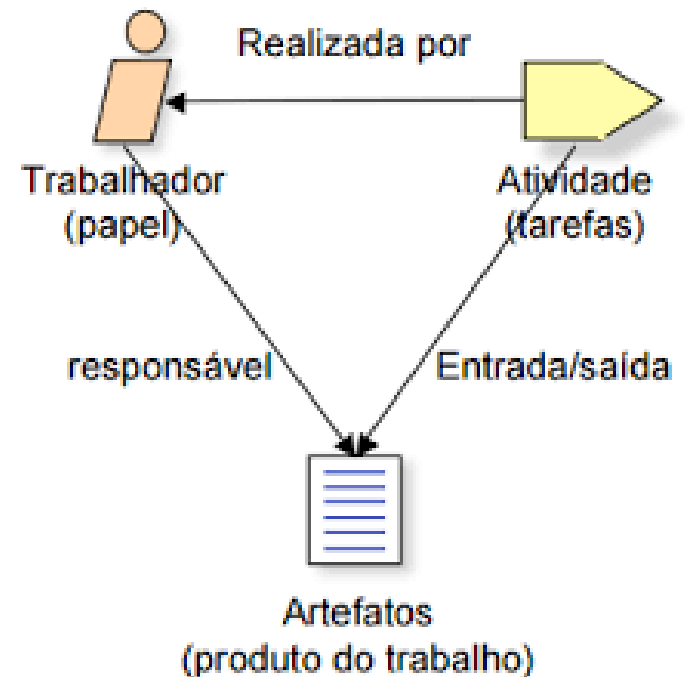


Pergunta

- Quais são as etapas da Engenharia de *Software*?
- Cria-se *software* para computadores da mesma forma que qualquer produto bem-sucedido: aplicando-se um processo adaptável e ágil que conduza a um resultado de alta qualidade, atendendo às necessidades daqueles que usarão o produto.

Pergunta

- O que é um Artefato na Engenharia de *Software*?



Pergunta

- O que é um Artefato na Engenharia de *Software*?
- Do ponto de vista de um engenheiro de *software*, artefato é um conjunto de programas, conteúdo, dados e outros artefatos que apoiam o *software* de computador;
- Do ponto de vista do usuário, o artefato é uma ferramenta ou um produto que, de alguma forma, torna a vida dele melhor.

Pergunta

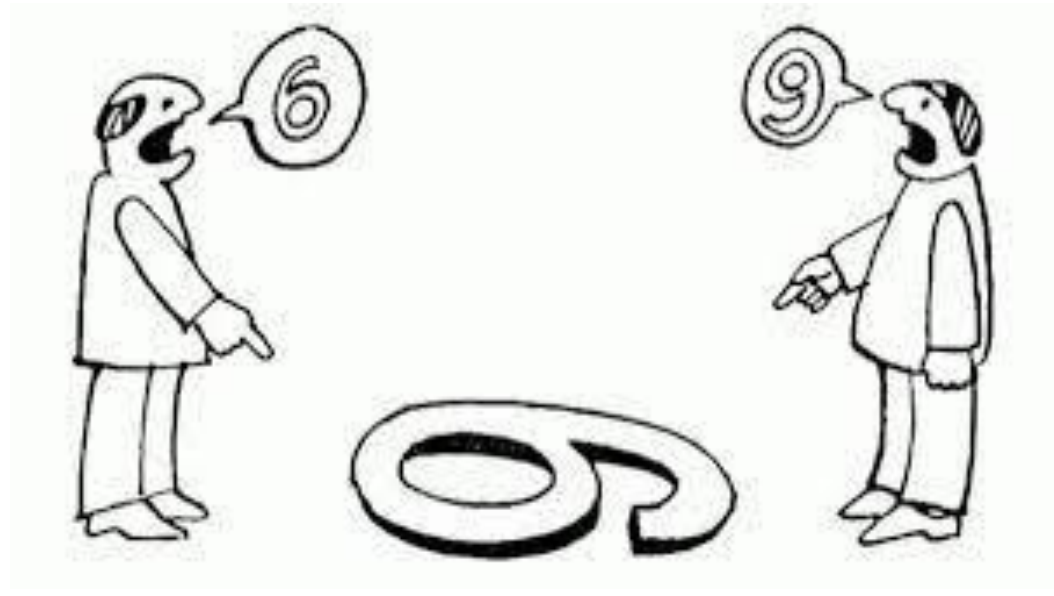
- Como garantir que o trabalho foi realizado corretamente?
- Definindo, elaborando e praticando os processos determinados pela Engenharia de *Software*.

ECM251 – Linguagens de Programação I

Aula 01 – L1/2 e L2/2

Pergunta

- O que é Paradigma?



Pergunta

- O que é Paradigma?
 - Define um exemplo típico, modelo de algo, ou uma forma de se entender alguma coisa;
 - É a representação de um padrão a ser seguido;
 - É uma realização científica com métodos e valores que são concebidos como modelo;
 - É uma referência inicial como base de modelo para estudos e pesquisas.

Pergunta

- O que é o Paradigma da Orientação a Objeto – OO?



Pergunta

- O que é o Paradigma da Orientação a Objeto – OO?
- Tenta gerenciar a complexidade dos problemas do mundo real abstraindo o conhecimento relevante e encapsulando-o em objetos;
- “Um sistema construído usando um método orientado a objetos é aquele cujos componentes são partes encapsuladas de dados e funções, que podem herdar atributos e comportamento de outros componentes da mesma natureza, e cujos componentes comunicam-se entre si por meio de mensagens” - Eduard Yourdon.

Pergunta

- O que é o Programação Orientada a Objetos – POO?



Pergunta

- O que é o Programação Orientada a Objetos – POO?
- Paradigma de programação baseado no conceito de objetos, que podem conter dados na forma de campos, também conhecidos como atributos, e códigos, na forma de procedimentos, também conhecidos como métodos;
- Uma característica de POO é que um procedimento de objeto pode acessar, e geralmente modificar, os campos de dados do objeto com o qual eles estão associados;
- Onde programas de computadores são projetados por meio da composição de objetos que interagem com outros.

Atividade

- Em grupos, pesquisar e apresentar (5 min.) na próxima aula L1/2 e L2/2, o seguinte tema:

Programação Orientada a Objetos

- Conceito;
- Vantagens e desvantagens;
- Aplicação e utilização;
- Principais linguagens de programação que utilizam OO.

ECM251 – Linguagens de Programação I

Aula 01

FIM