

Programação Orientada a Objetos (PO24CP-4CP)

Aula #01 - Aula Inaugural

Profa. Luciene de Oliveira Marin

lucienemarin@utfpr.edu.br

lucienemarin@professores.utfpr.edu.br (google)

Contato:

- Profa. Luciene, email: lucienemarin@utfpr.edu.br,
lucienemarin@professores.utfpr.edu.br
Sala 105 - Bloco S
- Site da disciplina no **Moodle UTFPR**:
 - <https://moodle.utfpr.edu.br/course/view.php?id=14146>
 - Senha de inscrição: **PO24CP-2023-02**
- **Horários de atendimento a aluno (PALuno)**:
 - **Segundas-feiras (2T4)** - 15h50 às 16h40, **Local: S105**
 - **Quartas-feiras (4T4, 4T5)** - 15h50 às 17h30, **Local: S105**
 - **Quintas-feiras (5T4)** - 15h50 às 16h40, **Local: S105**

Turma 4CP - Grade de horário e local:

- Segundas-feiras (2T2, 2T3) - 13h50 às 15h30, **Sala V105**
- Quartas-feiras (4T2, 4T3) - 13h50 às 15h30, **Sala V105**

Turma 4CPE - Grade de horário e local:

- Terças-feiras (3M4, 3M5) - 10h20 às 12h00, **Sala V106**
- Quintas-feiras (5M2, 5M3) - 08h20 às 10h00, **Sala V106**

Cronograma - Turma 4CP

Dias letivos do semestre 2023-02:

- Agosto: 09, 14, 16, 21, 23, 28, 30
- Setembro: 04, 06, 11, 13, 18, 20, 25, 27
- Outubro: 02 (T01), 04, 09, 11, 16, 18, 23, 25, 30
- Novembro: 01, 06, 08, 13, 20, 22, 27, 29
- Dezembro: 04 (T02), 06, 11, 13 (Reav.), 18, 20

Datas de entrega Trabalhos Práticos e Lista de Exercício Avaliativa:

- T01 (peso 0,35): 02/10 (segunda-feira)
- T02 (peso 0,45): 04/12 (segunda-feira)
- LEA (peso 0,2): Datas ao longo do período

Data da Prova de Reavaliação:

- Quando? 13/12 (quarta-feira)

Cronograma - Turma 4CPE

Dias letivos do semestre 2023-02:

- Agosto: 10, 15, 17, 22, 24, 29, 31
- Setembro: 05, 12, 14, 19, 21, 26, 28
- Outubro: 03 (T01), 05, 10, 17, 19, 24, 26, 31
- Novembro: 07, 09, 14, 21, 23, 28, 30
- Dezembro: 05 (T02), 07, 12 (Reav.), 19

Datas de entrega Trabalhos Práticos e Lista de Exercício Avaliativa:

- T01 (peso 0,35): 03/10 (terça-feira)
- T02 (peso 0,45): 05/12 (terça-feira)
- LEA (peso 0,2): Datas ao longo do período

Data da Prova de Reavaliação:

- Quando? 12/12 (terça-feira)

Ementa

- Conceitos do **paradigma da programação orientada a objetos**:
- Introdução a Classes e Objetos;
- Encapsulamento, Método Construtor;
- Agregação e Composição de objetos;
- Herança;
- Polimorfismo;
- Tratamento de Exceções;
- Coleções;
- Introdução a threads;
- Linguagem de programação orientada a objetos.

Objetivo

- Compreender os conceitos da programação orientação a objetos;
- Desenvolver aplicativos utilizando o paradigma de POO e uma linguagem de POO.

Carga horária em 2023-02 (Turma 4CP):

- Aulas previstas: 76 horas-aula;
- 25% = 19 horas-aula, 35% = 26,6 horas-aula e 50% = 38 horas-aula

Compensação de faltas (Res. COGEP/UTFPR nº110 -19/10/2021)

- Faltas $> 25\%$ e Faltas $\leq 35\%$ serão compensadas se nota final $\geq 6,5$
- Faltas $> 35\%$ e Faltas $\leq 50\%$ serão compensadas se nota final $\geq 8,0$

Procedimentos de Avaliação 1/2

Avaliações:

- 1 $T01$ e $T02$: trabalhos práticos de programação;
- 2 $MLEA$: média aritmética de listas de exercícios avaliativas.

Cálculo da Nota Final (NF)

$$NF = [0,35 * T01 + 0,45 * T02 + 0,2 * MLEA]$$

Aprovação:

- $NF \geq 6,0$ e frequência $\geq 75\%$ **ou**
- $NF \geq 6,5$ e frequência $\geq 65\%$ **ou**
- $NF \geq 8,0$ e frequência $\geq 50\%$

Prova de re-avaliação ($REAV$).

- Possibilidade **apenas** aos estudantes que não obtiverem média para aprovação.
- **Média final** = $[0,4 \times NF + 0,6 \times REAV]$

Observação 1:

- Todas as avaliações são cumulativas, de modo que o conteúdo abordado em avaliações anteriores poderá ser requerido nas avaliações seguintes.

Observação 2:

- O cronograma de atividades e das avaliações poderá sofrer alterações conforme necessidades e/ou eventualidades surgidas durante o desenvolvimento da disciplina.

Referência utilizada (8 exemplares na B.U.):

- Santos, Rafael. **Introdução a Programação Orientada a Objetos Usando Java**. 2. ed. Rio de Janeiro, RJ: Elsevier: Campus, 2013.

Referências Básicas:

- SEBESTA, R. W. **Conceitos de Linguagem de Programação**. 5ª Ed. Porto Alegre: Bookman, 2003.
- GOMES, E. B.
Dante Explica Java v.5. Rio de Janeiro: Ciência Moderna, 2005.
- SIERRA, Kathy. BATES, Bert.
Use a cabeça!: Java. 2. ed. Rio de Janeiro, RJ: Alta Books, 2007.

Referências Complementares:

- HORSTMANN, C. S. & CORNELL, G. **Core Java 2 - Volume 1 - Fundamentos**. 1ª Ed. Rio de Janeiro: Makron Books, 2000.
- **entre outras (vide plano de ensino oficial)**

Programação Estruturada - Revisão

Programação Estruturada:

O que caracteriza a programação estruturada?

- É uma forma de programar computadores estabelecida de forma que todos os programas possíveis possam ser reduzidos a apenas 3 estruturas:
 - **Sequência, decisão e iteração**

Como surgiu?

- Foi desenvolvida por Michael A. Jackson em seu livro "*Principles of Program Design*" em 1975

Por que a programação estruturada é importante? (1/2)

- 1) Orienta aos programadores a criação de estruturas simples em seus programas:
 - **utilize subrotinas e funções.**
- 2) Foi a forma dominante na criação de software anterior à **programação orientada a objetos (POO)**.
 - Apesar de ter sido sucedida pela POO, a programação estruturada ainda é muito influente.
Por que? Grande parte das pessoas ainda aprendem programação através dela.

Por que a programação estruturada é importante? (2/2)

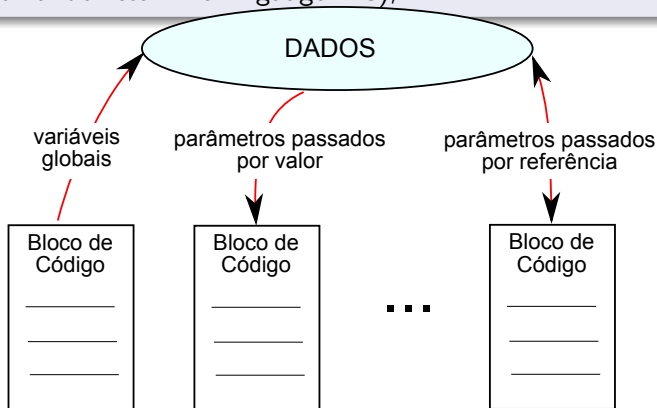
- 3) Para a resolução de problemas relativamente mais simples e diretos:
 - **A programação estruturada é muito eficiente**
- 4) Por exigir formas de pensar relativamente complexas, a POO até hoje ainda não é bem compreendida ou usada pela maioria.
- 5) Inúmeras linguagens ainda extremamente relevantes nos dias de hoje, como Cobol, PHP e Perl ainda utilizam o paradigma estruturado (muito embora possuam suporte para a orientação à objetos).

Principais características da Programação Estruturada: (1/2)

- programação orientada a procedimentos: subprogramas = blocos estruturados de códigos (procedimentos, funções ou módulos);
- a comunicação entre os blocos se faz utilizando variáveis globais e pela passagem de dados através de parâmetros;
- a execução de um programa é caracterizada pelo acionamento de um bloco de código.
- Obs.: a utilização de variáveis globais não constitui uma boa prática de programação (escopo muito grande).

Principais características da Programação Estruturada: (2/2)

- os dados são processados nos blocos e migram de um bloco para outro, através de variáveis globais, parâmetros passados por referência e expressão retornada pela função (através do comando return na linguagem C);



Lista de Exercícios:

- Revisão da Linguagem C e Algoritmos e Estrutura de Dados 1
 - [Vide moodle da disciplina.](#)