

MC302 - PROGRAMAÇÃO ORIENTADA A OBJETOS

- Período: Primeiro Semestre de 2018
- Turmas: E e F
- Docente Responsável: Prof. Julio Cesar dos Reis
- Sala: 74 (IC-02)
- E-Mail: jreis@ic.unicamp.br
- Página: <http://www.ic.unicamp.br/~jreis>
- Website do curso: <http://googleapps.unicamp.br> – Código da turma: qr97v7

1 Aulas e Atendimento

As aulas teóricas serão ministradas nos seguintes dias e horários:

- Terças-feiras das 21:00 às 23:00 na sala a ser divulgada.
- Quintas-feiras das 19:00 às 21:00 na sala a ser divulgada.

As aulas de laboratório serão nos seguintes dias e horários:

- Sábados das 10:00 às 12:00 nas salas 302 e 303 (IC-03).

O horário de atendimento extra-classe será realizado sempre depois das aulas às quintas-feiras ou com agendamento prévio com o Professor na Sala 74 (IC-02).

2 Programa da Disciplina

Ao final deste curso o(a) aluno(a) deverá ser capazes de implementar adequadamente sistemas de software usando uma linguagem orientada a objetos (OO). O curso exercitará conceitos básicos e avançados em modelo OO. O(a) aluno(a) desenvolverá programas usando técnicas de programação OO em aplicações específicas. Os seguintes tópicos serão tratados:

- Conceitos básicos em OO: objeto, classe, interface pública, estado, mensagens, encapsulamento de dados e operações, comportamento
- Modelagem OO: UML
- Introdução ao Java: tipos primitivos e seus operadores, estruturas de controle
- Classes e Objetos: declaração de classes, métodos e variáveis, instanciação de objetos, construtores, sobrecarga de métodos, visibilidade de métodos e atributos
- Métodos e atributos: métodos e atributos estáticos, variáveis e métodos finais
- Vetores estáticos e dinâmicos: Arranjos unidimensionais e multidimensionais, classe *Array* e *ArrayList*
- Enumeração
- Classes Aninhadas
- Relacionamentos entre classes: Hierarquias de generalização/especialização, Hierarquias de agregação/decomposição, Associação entre classes
- Herança Simples e Múltiplas
- Polimorfismo: polimorfismo paramétrico e de inclusão, sobrecarga de métodos, redefinição de operações, acomplamento estático e dinâmico
- Classes abstratas, Interfaces e Modularização
- Classes e coleções genéricas
- Metaclasses
- Tratamento de Exceções
- Entrada e Saída: serialização e persistência de objetos
- Programação concorrente (threads)
- Interfaces Gráficas
- Introdução aos Padrões de Projeto

Linguagens de Programação. A linguagem de programação *Java* é adotada no curso. As salas de laboratórios alocados disponibilizam máquinas com ambientes de desenvolvimento para essa linguagem.

3 Listas de Exercícios

Ao longo do curso serão publicados exercícios de fixação que não precisarão ser entregues, mas recomenda-se fortemente a resolução dos mesmos. Haverá resolução de exercícios em sala que poderão contribuir na nota final dos testes teóricos.

4 Testes Teóricos

Serão aplicados t testes com k questões, sendo $k \leq 2$, ao longo do semestre. A duração dos testes será de 40min no final do horário de aulas teóricas, em datas a serem divulgadas em avisos na página do curso. Seja T_i a nota do teste no intervalo $[0, 10]$, M_T é a média aritmética dos testes.

$$M_T = \frac{T_1 + \dots + T_t}{t}.$$

5 Laboratórios

Serão realizados n laboratórios (trabalhos de programação) **individualmente** ao longo do semestre. O prazo de entrega dos laboratórios será compatível com o nível de dificuldade segundo especificado no enunciado do trabalho.

Os trabalhos deverão ser elaborados usando a linguagem *Java* e submetidos via o ambiente de ensino-aprendizado. Na correção dos laboratórios, serão considerados os seguintes critérios básicos:

- aderência ao enunciado
- corretude da solução proposta
- conceitos OO aplicados e implementados adequadamente
- organização e qualidade do código desenvolvido

Critérios específicos podem ser estabelecidos no enunciado do laboratório. Os monitores entregarão as correções e comentários sobre o programa se requerido pelo aluno.

Cada laboratório terá um peso $LP_i \in \{1, 2, 3\}$. Seja L_i a nota do trabalho de programação no intervalo $[0,10]$ e M_L a média ponderada dos n laboratórios.

$$M_L = \frac{LP_1 * L_1 + \dots + LP_n * L_n}{LP_1 + \dots + LP_n}.$$

6 Projeto Computacional

Um projeto prático deverá ser desenvolvido visando a aplicação transversal dos conceitos de programação OO estudados ao longo do semestre. Esse projeto deve ser realizado em grupo de 4 a 5 alunos. Além da especificação e código resultante do projeto, o grupo deverá apresentar um relatório que descreva o projeto desenvolvido. Uma especificação detalhada do enunciado desta atividade será publicada pelo docente. Haverão entregas parciais de partes do projeto ao longo do semestre e as datas serão divulgadas na página do curso. As entregas parciais deverão ser respeitadas. Considere M_P a nota do projeto final de programação no intervalo $[0,10]$.

7 Atividade de Aprendizagem Ativa

Uma atividade de pesquisa e síntese deverá ser elaborada e apresentada pelos alunos. Essa atividade deve ser realizado em grupo de 4 a 5 alunos. O docente irá propor uma série de temas, sugerir bibliografias e conduzir reunião de acompanhamento com os grupos. Os alunos deverão pesquisar e organizar materiais, efetuarão leituras e desenvolverão discussão em grupo. Os alunos deverão elaborar um seminário para expor os resultados sobre o tema pesquisado. O docente irá promover discussão entre os alunos em sala de aula sobre o tema e os conceitos em estudo. Uma especificação detalhada do enunciado desta atividade será publicada na página do curso. Considere M_S a nota da atividade de aprendizagem ativa no intervalo $[0,10]$.

8 Critérios de Aprovação

- A média M , antes do exame, será calculada da seguinte maneira:

$$M = \begin{cases} 0 & \text{se } M_T = 0 \text{ e } M_P = 0 \\ 0.25M_T + 0.25M_L + 0.40M_P + 0.10M_S & \text{caso contrário} \end{cases}$$

- Aluno com média $2,5 \leq M < 5,0$ poderá fazer um exame final (seja E a nota do exame).
- O exame consistirá de uma prova teórica escrita contendo j questões com o conteúdo do curso todo. A nota E será no intervalo $[0, 10]$.
- A nota final, F , será calculada como:

$$F = \begin{cases} \min\{5,0; \frac{M + E}{2}\} & \text{caso } 2,5 \leq M < 5,0 \text{ e o aluno tenha realizado o exame} \\ M & \text{caso contrário} \end{cases}$$

- O aluno estará aprovado caso sua nota final F seja maior ou igual a 5,0, e tiver pelo menos 75% de frequência nas aulas. Estará reprovado caso contrário.

Data do exame

- 10/Julho/2018

Observações

- Não haverá testes ou laboratórios substitutivos.
- **Qualquer tipo de fraude acarretará em nota final $F = 0$ (zero) para todos os envolvidos, sem prejuízo de outras sanções.**

9 Monitoria

Os PEDs e PADs estarão disponíveis para atendimento nos seguintes dias e horários:

- Terças-feiras das 18:00 às 19:00 na sala 302 (IC-03)
- Quintas-feiras das 18:00 às 19:00 na sala 302 (IC-03)

Monitores:

- Alexandre Rodrigues Gomes (PED)
- Italos Estilon da Silva de Souza (PED)
- Guilherme Mendeleh Perrotta (PAD)

10 Referências

Os seguintes livros cobrem o conteúdo proposto no curso. Materiais complementares e de apoio serão indicados ao longo do curso:

1. Java: Como Programar, Paul Deitel & Heivey Deitel; Pearson; 7a. Ed.
2. Java in a Nutshell, David Flanagan, O'Reilley, 2005.
3. The Java Programming Language, Ken Arnold, James Gosling, & David Holmes; Prentice Hall, 4th edition (2005)
4. Thinking in Java, Bruce Eckel; Prentice Hall, 2th edition (2000)
5. Data Structures and Algorithms with Object Oriented Design Patterns in Java, Bruno Preiss;
6. The Java Tutorials (Oracle)
(<http://docs.oracle.com/javase/tutorial/>)
7. Guia do Usuário UML, Grady Booch *et al.*; Campus(1999)
8. Java Pocket Guide - Robert Liguori & Patricia Liguori; O'Reilley, 2008.