



## Tecnologia em Análise e Desenvolvimento de Sistemas

Componente Curricular: <b>Programação II</b>				
Professor Moser Silva Fagundes				
Período Letivo 2018/2	Período Semestre	Turma ADS.2N	Carga Horária Total 90 h	Total de Horas-Aula 120 h
Horário de Aula: Terças-feiras das 20 h 45 min às 22 h 15 min; quintas-feiras das 19 h às 22 h 15 min				
Horário de estudos orientados: Quintas-feiras das 17 h às 19 h				

### EMENTA DO COMPONENTE CURRICULAR

Introdução ao paradigma de Orientação a Objetos e compreensão dos conceitos fundamentais: herança, polimorfismo, interface, encapsulamento de dados, sobrecarga e sobrescrita de métodos. Utilização de linguagem orientada a objetos. Tratamento de exceções e erros. Estudo de tipos de estruturas de dados: listas lineares (pilhas e filas); e listas encadeadas. Compreensão e aplicação de persistência de dados.

#### OBJETIVO GERAL

Compreender os conceitos fundamentais do paradigma de programação orientado a objetos envolvidos no desenvolvimento de sistemas computacionais.

#### Objetivos Específicos

- Aplicar técnicas de programação orientada a objetos;
- Implementar as principais estruturas de dados;
- Desenvolver programas computacionais orientados a objeto como forma de resolução de problemas.

### Conteúdos programáticos

1. Fundamentos da linguagem Java
  - 1.1. Máquina Virtual
  - 1.2. Conceitos básicos da linguagem Java
  - 1.3. Operadores
  - 1.4. Estruturas de controle
  - 1.5. Métodos
  - 1.6. Coleções
2. Paradigma da Programação Orientada a Objetos
  - 2.1. Conceitos fundamentais (classes, objetos, métodos e atributos)
  - 2.2. Construtores
  - 2.3. Encapsulamento de dados
  - 2.4. Sobrecarga e sobrescrita de métodos
  - 2.5. Relacionamentos entre classes (associação, agregação e composição)
  - 2.6. Herança
  - 2.7. Polimorfismo
  - 2.8. Interface
3. Tratamento de exceções e erros
  - 3.1. Captura de exceções
  - 3.2. Delegação do tratamento de exceções
4. Persistência de dados

- 5. Desenvolvimento de Interface Gráfica
  - 5.1. Componentes Swing
  - 5.2. Gerenciadores de layout
- 6. Estruturas de dados
  - 6.1. Listas
  - 6.2. Filas
  - 6.3. Pilhas

### **Metodologia de ensino**

Aulas expositivas e práticas utilizando quadro branco e exibição de apresentações gráficas, bem como software disponível no laboratório de informática; Resolução de exercícios sobre os conteúdos abordados; Discussões em grupo; Pesquisas para ampliar o conteúdo apresentado em aula.

### **Critérios e metodologia de avaliação do processo de ensino-aprendizagem**

Os critérios para avaliação dos trabalhos são:

- (a) Adequação aos objetivos do trabalho e objetividade das respostas;
- (b) Linguagem escrita (português);
- (c) Pontualidade na entrega;
- (d) Aplicação adequada dos conhecimentos adquiridos na disciplina.

No caso dos trabalhos práticos, também serão usados como critérios de avaliação:

- (a) Complexidade do projeto;
- (b) Corretude do código levando em conta o seu propósito;
- (c) Compreensão do tema proposto no exercício;
- (d) Participação na resolução do exercício.

As notas serão compostas da seguinte forma:

- Nota 1 (30 pontos): Avaliação AV1 - Lista de exercícios: prática e/ou teórica.
- Nota 2 (30 pontos): Avaliação AV2 - Prova: avaliação prática e/ou teórica.
- Nota 3 (40 pontos): Avaliação AV3 - Trabalhos prático: trabalho individual abrangendo todo o conteúdo visto na disciplina, incluindo estruturas de dados. O trabalho inclui descrição textual, implementação de código e apresentação para a turma.

Nota Semestral = Nota 1 + Nota 2 + Nota 3

Aprovação: o resultado da avaliação do desempenho do aluno na disciplina será expresso por meio da nota semestral. O aluno que obtiver a nota semestral igual ou superior a sessenta (60,0) na disciplina é considerado aprovado, desde que a frequência atenda o previsto em lei. Cada nota parcial é expressa em grau numérico, considerando-se uma casa decimal e utilizando-se da regra de arredondamento matemático. A nota final será expressa em números inteiros.

Com objetivo de recuperar qualitativa e quantitativamente os conteúdos e práticas, a realização de estudos de recuperação se dará em horário de estudos orientados.

**Mensagem automática 1:** observar se nessa seção está prevista a Recuperação Paralela. Ela não é uma “prova substitutiva”, pois envolve a oferta de oportunidades para alunos de aprendizagem (inclusive para alunos com bom desempenho). Deve ser algum texto tal como: “Paralelamente às aulas, serão desenvolvidas atividades de recuperação aos alunos nos horários de estudos orientados, principalmente para alunos de baixo rendimento, nos termos da Seção I, Subseção V, da Organização Didática (art. 205) e Parecer CNE/CEB 12/97.”

**Mensagem automática 2** (somente para Licenciaturas): Sua disciplina prevê “práticas de ensino”? Elas são momentos que devem aproximar ao máximo os estudantes das situações cotidianas de sala de aula e da vivência que terão como professores. Atendendo à carga horária prevista, devem propor atividades que os auxiliem, por exemplo, na atuação durante os estágios. Então, cada professor tem autonomia para propor as atividades que considerar mais adequadas, como: construção de materiais didáticos, fóruns, observações de situações de sala de aula, jogos, elaboração de planos de aula... Lembre-se que é preciso converter as horas previstas nas matrizes curriculares em hora-aula. As práticas de ensino não precisam estar concentradas em dias específicos, mas podem estar distribuídas ao longo do semestre.

**Bibliografia básica**

DEITEL, H.M.; DEITEL, P.J. Java como programar. 8. ed. Porto Alegre: Bookman, 2010.

BARNES, D.; KOLLING, M. Programação orientada a objetos com Java: uma introdução prática usando o BlueJ. 4. ed. São Paulo: Pearson Prentice Hall, 2009.

PEREIRA, S. L. Estruturas de dados fundamentais: conceitos e aplicações. 12. ed. São Paulo: Érica, 2008.

**Bibliografia complementar**

WAZLAWICK, R. S. Análise e projetos de sistemas de informação orientados a objetos. 2. ed. São Paulo: Elsevier Campus, 2010.

SINTES, T. Aprenda programação orientada a objetos em 21 dias. São Paulo: Pearson Makron Books, 2010.

LAFORE, R. Estruturas de dados e algoritmos em Java. Rio de Janeiro: Ciência Moderna, 2005.

SANTOS, R. Introdução à programação orientada a objetos usando Java. São Paulo: Elsevier, 2003.

BARRY, P. Use a cabeça!: Java. Rio de Janeiro: Alta Books, 2010.

Formulário original em: 31/07/2018 11:48:25



## Tecnologia em Análise e Desenvolvimento de Sistemas

Componente Curricular: <b>Programação II</b>				
Professor Moser Silva Fagundes				
Período Letivo 2018/2	Período Semestre	Turma ADS.2N	Carga Horária Total 90 h	Total de Horas-Aula 120 h

Segue esboço de cronograma. É possível que seu calendário mostre algumas efemérides - datas comemorativas ou de profissão - apenas como referência para eventual atividade especial. Preencha conforme adequado e apague essa mensagem.

Data	Aulas	Dia da semana	Conteúdos
31/07/2018	2	Terça-Feira	Apresentação da disciplina Lista de exercícios de revisão dos conteúdos de Programação I
02/08/2018	4	Quinta-Feira	Revisão sobre variáveis, tipos de dados, laços de repetição, condicionais Instalação do ambiente Eclipse Importação e exportação de projetos
07/08/2018	2	Terça-Feira	Comandos break e continue Classe Math e classe Random
09/08/2018	4	Quinta-Feira	Arrays em Java Classe String
14/08/2018	2	Terça-Feira	Arrays multidimensionais
16/08/2018	4	Quinta-Feira	Introdução a orientação a objetos Classes, objetos, atributos e métodos
21/08/2018	2	Terça-Feira	Exercícios sobre construção de classes e objetos
23/08/2018	4	Quinta-Feira	Modificador de acesso private Métodos get e set Construtores
28/08/2018	2	Terça-Feira	Associação, agregação e composição
30/08/2018	4	Quinta-Feira	Arrays de objetos Aspectos avançados da classe String
01/09/2018	4	Sábado	Sábado com aulas de Quinta-feira : Mostra Cultural
04/09/2018	2	Terça-Feira	Atributos e métodos de classe (estáticos)

06/09/2018	4	Quinta-Feira	Herança de classes
11/09/2018	2	Terça-Feira	Exercícios de revisão de classes, objetos e herança
13/09/2018	4	Quinta-Feira	Avaliação AV1 (30 pontos)
18/09/2018	4	Terça-feira	Terça-feira com aulas de Quinta-feira : Letivo (Semestrais) Sobrescrita e sobrecarga de métodos Polimorfismo
20/09/2018	0	Quinta-Feira	Data Estadual
25/09/2018	2	Terça-Feira	Tratamento de exceções com try catch
27/09/2018	4	Quinta-Feira	Classes abstratas Polimorfismo (continuação)
02/10/2018	2	Terça-Feira	Interfaces
04/10/2018	4	Quinta-Feira	Classe ArrayList Outras coleções de linguagem Java
09/10/2018	2	Terça-Feira	Exercícios sobre coleções
11/10/2018	4	Quinta-Feira	Arquivos
16/10/2018	2	Terça-Feira	Arquivos
18/10/2018	4	Quinta-Feira	Exercícios de arquivos
23/10/2018	2	Terça-Feira	Exercícios envolvendo coleções e arquivos
25/10/2018	4	Quinta-feira	Mostra técnica
30/10/2018	2	Terça-Feira	Exercícios de revisão para a prova sobre herança, sobrescrita, sobrecarga, polimorfismo, coleções, arquivos
01/11/2018	4	Quinta-Feira	Avaliação AV2 (30 pontos)
06/11/2018	2	Terça-Feira	Construção de GUI no WindowBuilder, layouts e componentes Divulgação do trabalho prático AV3-1 (30 pontos)
08/11/2018	4	Quinta-Feira	Criação de múltiplas telas no WindowBuilder
13/11/2018	2	Terça-Feira	Layouts, Componentes, Menus, Imagens
15/11/2018	0	Quinta-Feira	Data Nacional
20/11/2018	2	Terça-feira	Terça-feira com aulas de Terça-feira : [Dia da Consciência Negra] Estruturas de Dados (ED) - Conceitos Gerais
22/11/2018	4	Quinta-Feira	Estruturas de Dados (ED) - Listas Lineares / Buscas Básicas
27/11/2018	2	Terça-Feira	Estruturas de Dados (ED) - Listas Encadeadas

29/11/2018	4	Quinta-Feira	Estruturas de Dados (ED) - Filas e Pilhas
04/12/2018	2	Terça-Feira	Desenvolvimento do trabalho prático AV3-1
06/12/2018	4	Quinta-Feira	AV3-2 Trabalho Prático de ED (10 pontos)
11/12/2018	2	Terça-Feira	Apresentação do trabalho AV3-1 (30 pontos)
13/12/2018	4	Quinta-Feira	Apresentação do trabalho AV3-1 (30 pontos) (continuação)
18/12/2018	2	Terça-Feira	Encerramento da Disciplina
Total de aulas	120		

em análise por Carla em [[DATA]].