

CIN 7907 – PLANO DE ENSINO (2017.2)

1 IDENTIFICAÇÃO

DISCIPLINA: CIN 7907 – Lógica Aplicada I

CARGA HORÁRIA: 72h semestrais / 4h semanais

PROFESSOR: Moisés Lima Dutra (moises.dutra@ufsc.br)

OFERTA: Optativa para os Cursos de Ciência da Informação, Arquivologia, Biblioteconomia e áreas afins

HORA E LOCAL: Quintas-feiras, das 14h20 às 18h00, LABINFOR

2 EMENTA

Modelagem de sistemas básicos. Estruturas de dados simples. Algoritmos. Orientação a objetos.

3 OBJETIVOS

3.1 Objetivo Geral

Desenvolver a capacidade de resolução de problemas lógicos de complexidade básica por meio da construção e aplicação de algoritmos.

3.2 Objetivos Específicos

3.2.1. Identificar elementos de problemas lógicos.

3.2.2. Modelar cenários de aplicação orientados a objetos.

3.2.3. Modelar estruturas de dados de suporte a algoritmos.

3.2.4. Implementar soluções algorítmicas a partir do paradigma da orientação a objetos.

4 CONTEÚDO PROGRAMÁTICO

4.1 Máquina de Turing

4.2 Arquiteturas Lógicas de Processamento Computacional

4.2.1 Arquitetura de John Von Neumann

4.2.1.1 Unidade de Controle (UC)

4.2.1.2 Memória

4.2.1.3 Unidade Lógica e Aritmética (ULA)

4.2.1.4 Dispositivos de Entrada e Saída de Dados (E/S)

4.2.2 Arquitetura de Harvard

4.2.3 Outras Arquiteturas

4.3 Algoritmos

4.3.1 Histórico

4.3.2 Tipos de Dados

4.3.3 Variáveis Simples

4.3.4 Variáveis Compostas

4.3.5 Procedimentos

- 4.3.6 Funções
- 4.3.7 Condições
- 4.3.8 Laços de Repetição

4.4 Orientação a Objetos

- 4.4.1 Objeto
- 4.4.2 Atributo
- 4.4.3 Método
- 4.4.4 Classe
- 4.4.5 Herança
- 4.4.6 Polimorfismo

4.5 Resolução de Problemas Lógicos via Algoritmos

5 METODOLOGIA

Aulas expositivas e atividades práticas de laboratório. Resoluções de problemas propostos via algoritmos e o desenvolvimento de um projeto final de semestre complementam o conjunto de ferramentas com as quais os alunos trabalharão os conhecimentos desenvolvidos durante a disciplina.

O Moodle (<http://moodle.ufsc.br/>) será utilizado como instrumento de suporte da disciplina, devendo o mesmo, portanto, ser **consultado todas as semanas** pelos alunos.

6 AVALIAÇÃO

Conforme a Resolução do Conselho Universitário 017/Cun/97, de 30/09/97:

- ❖ A frequência mínima obrigatória é de 75% das aulas.
- ❖ **Em caso de falta em dia de avaliação (prova, apresentação de trabalho ou mini-seminário)**, encaminhar justificativa e pedido formal à Chefia do Departamento de Ciência da Informação, no prazo de três (3) dias úteis.

Avaliações:

- Serão realizadas através de uma prova individual (**PE**), de atividades práticas de desenvolvimento de algoritmos (**AP**) e de um projeto final de semestre (**PS**).

A média final (**MF**) será calculada conforme a seguinte fórmula:

$$\mathbf{MF = 0.4(PE) + 0.2(AP) + 0.4(PS)}$$

- Arredondamentos de notas só serão feitos com a Média Final (**MF**).
- ❖ O aluno que obtiver a **MF entre 3,0 e 5,5** e frequência suficiente poderá, ao final do semestre, realizar uma prova de recuperação de todo o conteúdo.
- ❖ **EM CASO DE RECUPERAÇÃO**, a nota final do semestre (**NFS**) será calculada a partir da **média simples** entre a média final (**MF**) obtida durante o semestre e a nota obtida na prova de recuperação (**PR**).

$$\mathbf{NFS = (MF + PR) / 2}$$

7 BIBLIOGRAFIA

Básica:

ARAÚJO, Everton Coimbra de. **Algoritmos: fundamento e prática**. 3. ed. ampl. e atual. Florianópolis: Visual Books, 2007. 414 p. ISBN 9788575022092.

GUIMARÃES, Angelo de Moura; LAGES, Newton Alberto de Castilho. **Algoritmos e estruturas de dados**. Rio de Janeiro: Livros Tecnicos e Cientificos, 1985. 216p. ISBN 8521603789.

PREISS, Bruno R. **Estruturas de dados e algoritmos: padroes de projetos orientados a objetos com java**. Rio de Janeiro: Campus, c2001. xvi, 566p. ISBN 8535206930.

Complementar:

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores: algoritmos, Pascal, C/C++ e Java**. 2. ed. São Paulo: Pearson Prentice Hall, 2008. 434p. ISBN 9788576051480.

BABIN, Lee. **Beginning Ajax with PHP: From Novice to Professional**. Berkeley, CA: Apress, Inc., 2007. ISBN 9781430202530 Disponível em: <<http://dx.doi.org/10.1007/978-1-4302-0253-0>>. Acesso em: 02 jun. 2016.

BOOCH, Grady et al. **Object-oriented analysis and design with applications**. 3rd ed. Upper Saddle River: Addison Wesley, c2007. xxiii, 691 p. ISBN 9780201895513.

BLAHA, Michael; RUMBAUGH, James. **Modelagem e projetos baseados em objetos com UML 2**. 2. ed. rev. e atual. Rio de Janeiro: Elsevier, 2006. xvii, 496 p. ISBN 8535217533.

CADENHEAD, Rogers. **Aprenda em 24 horas Java 1.1**. Rio de Janeiro: Campus, 1998. xxi, 379 p. (Aprenda). ISBN 8535202064.

CARRANO, Frank M. **Data abstraction and problem solving with C++: walls and mirrors**. 4th. ed. Boston: Addison Wesley, c2005. xxiv, 968 p. ISBN 0321247256.

FRAIZER, Colin; BOND, Jill. **API JAVA: manual de referência**. São Paulo: Makron Books, c1997. 371p ISBN 8534607583.

GOLDMAN, Alfredo; KON, Fabio; SILVA, Paulo J.S. **Introdução à ciência de computação com Java e orientação a objetos**. São Paulo:IME/USP, 2004. 1.ed. 192 p. ISBN: 8588697106. Disponível em: <<http://ccsl.ime.usp.br/files/books/intro-java-cc.pdf>>. Acesso em: 02 jun. 2016.

GOODRICH, Michael T.; TAMASSIA, Roberto. **Data structures and algorithms in java**. 5th ed. New York: John Wiley & Sons, 2010 714p. ISBN 9780470383261.

GROSS, Hans-Gerhard. **Component-Based Software Testing with UML**. Berlin: Springer-Verlag Berlin Heidelberg, 2005.

HEEGE, Marcus. **Expert C++/CLI: .NET for Visual C++ Programmers**. Berkeley, CA: Apress, Inc., 2007. ISBN 9781430203575 Disponível em: <<http://dx.doi.org/10.1007/978-1-4302-0357-5>>. Acesso em: 02 jun. 2016.

HETLAND, Magnus Lie. **Beginning Python: From Novice to Professional**. Berkeley, CA: Apress, Inc., 2005. ISBN 9781430200727 Disponível em: <<http://dx.doi.org/10.1007/978-1-4302-0072-7>>. Acesso em: 02 jun.2016.

LANGTANGEN, Hans Petter. **Python Scripting for Computational Science**. Third Edition. Berlin: Springer-Verlag Berlin Heidelberg, 2008. (Texts in Computational Science and Engineering, 1611-0994; 3).

LARMAN, Craig. **Applying UML and patterns: an introduction to object-oriented analysis and design and the unified process**. 2nd ed. Upper Saddle River: Prentice Hall PTR, c2002. xxi, 627 p. ISBN 0130925691.

LIBERTY, Jesse. **Aprenda em 24 horas C++**. 3. ed. Rio de Janeiro: Campus, 1998. 489p. ISBN 8535202358.

MEDEIROS, Ernani Sales de. **Desenvolvendo software com UML 2.0: definitivo**. São Paulo: Pearson Makron Books, 2004. 264 p. ISBN 8534615292.

ODELL, James J. **Advanced object-oriented analysis and design using UML**. Cambridge, MA: Cambridge University Press, New York: Sigs Books, 1998. 246 p. ISBN 052164819X.

POWERS, David. **PHP Object-Oriented Solutions**. Berkeley, CA: Apress, Inc, 2008. ISBN 9781430210122
Disponível em: <<http://dx.doi.org/10.1007/978-1-4302-1012-2>>. Acesso em: 02 jun. 2016.

PURVIS, Michael; SAMBELLS, Jeffrey; TURNER, Cameron. **Beginning Google Maps Applications with PHP and Ajax: From Novice to Professional**. Berkeley, CA: Apress, Inc., 2006. ISBN 9781430202240
Disponível em: <<http://dx.doi.org/10.1007/978-1-4302-0224-0>>. Acesso em: 02 jun.2016.

RANGEL, Ricardo. **Curso prático: programação orientada a objetos: com Borland C++ para Windows**. São José dos Campos: Rangel, 1998. 473p.

SILVA, Ricardo Pereira e. **Como modelar com UML 2**. Florianópolis: Visual Books, 2009. 319p. ISBN 9788575022436

STROUSTRUP, Bjarne. **A linguagem de programação C++**. 3. ed. Porto Alegre: Bookman, 2000. 823p ISBN 8573076992.

TITTEL, Ed; GAITHER, Mark. **60 minutos para aprender Java**. São Paulo Berkeley, 1996. xv, 229p ISBN 8572513922.

WAZLAWICK, Raul Sidnei. **Análise e design orientados a objetos para sistemas de informação: modelagem com UML, OCL e IFML**. 3. ed. Rio de Janeiro: Elsevier: Campus, c2015. 462 p. ISBN 9788535279849.

8 CRONOGRAMA *		
AULA	DATA	CONTEÚDO
1	03/08	Apresentação da disciplina.
1	10/08	Algoritmos.
3	17/08	Algoritmos.
4	24/08	Algoritmos.
5	31/08	Algoritmos.
6	07/09	Atividade Extraclasse.
7	14/09	Desenvolvimento Orientado a Objetos.
8	21/09	Desenvolvimento Orientado a Objetos.
9	28/09	Desenvolvimento dos Projetos Finais.
10	05/10	Desenvolvimento dos Projetos Finais.
11	12/10	Atividade Extraclasse!
12	19/10	Prova Individual.
13	26/10	Atividade Extraclasse!
14	02/11	Atividade Extraclasse!
15	09/11	Desenvolvimento dos Projetos Finais.
16	16/11	Desenvolvimento dos Projetos Finais.
17	23/11	Apresentação dos Projetos Finais.
18	30/11	Prova de Recuperação.

**Cronograma sujeito a alterações e adaptações ao longo do semestre.*