



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS ARARANGUÁ-ARA

PLANO DE ENSINO

SEMESTRE 2014.2

**I. IDENTIFICAÇÃO DA DISCIPLINA:**

CÓDIGO	NOME DA DISCIPLINA	Nº DE HORAS-AULA SEMANAIS		TOTAL DE HORAS-AULA SEMESTRAIS
		TEÓRICAS	PRÁTICAS	
ARA7143	Lógica de Programação	2	2	72

HORÁRIO		MÓDULO
TURMAS TEÓRICAS	TURMAS PRÁTICAS	Presencial
01655 – 2-2020-2 – 6-1620-2	01655 – 2-2020-2 – 6-1620-2	

**II. PROFESSOR(ES) MINISTRANTE(S)**

Jim Lau  
e-mail: jimlau.am@gmail.com

**III. PRÉ-REQUISITO(S)\***

CÓDIGO	NOME DA DISCIPLINA

**IV. CURSO(S) PARA O(S) QUAL(IS) A DISCIPLINA É OFERECIDA**

Graduação em Engenharia de Computação

**V. JUSTIFICATIVA**

O profissional responsável por analisar e projetar sistemas computacionais necessita conhecer e aplicar as principais metodologias adotadas pelo mercado de trabalho, para poder desempenhar sua função com qualidade e ser competitivo no mercado.

**VI. EMENTA**

Lógica de Programação. Sequências lógicas, pseudocódigo, fluxograma, diagrama de chapin. Variáveis: nomeação, declaração, inicialização, tipos de dados. Expressões aritméticas, expressões literais, expressões lógicas, expressões relacionais. Arquitetura de um programa Mínimo: paradigma estruturado, regras de escopo, funções, modularização. Estruturas de Dados Simples: vetores, matrizes, registros. Estruturas de Controle de Fluxo: Linear, condicional, repetição. Entrada e Saída de Dados: noções de arquivo, acesso sequencial, acesso direto.

**VII. OBJETIVOS**

**Objetivos Gerais:**

- ✓ O aluno ao final desta disciplina deverá ser capaz de transpor para a forma algorítmica soluções de problemas, utilizando-se de notações formais de representação de algoritmos, tais como, pseudo-linguagens e diagramas de fluxo.

**Objetivos Específicos:**

Desenvolver o raciocínio lógico e abstrato no contexto computacional, visando estruturar os pensamentos de maneira sistêmica para entender a programação de computadores, bem como saber resolver problemas e automatizar tarefas e, para que isso seja possível ao término da disciplina o aluno deverá:

- ✓ Conhecer os componentes computacionais e os conceitos relevantes a área da computação, no que tange a lógica de programação;
- ✓ Saber construir, interpretar e testar algoritmos para a resolução de problemas.
- ✓ Compreender as definições de constantes e variáveis e aplicá-las no desenvolvimento de algoritmos;

- ✓ Desenvolver expressões aritméticas, literais, lógicas e relacionais e aplicá-las na resolução de problemas através do uso de algoritmos;
- ✓ Saber utilizar as estruturas de controle: condicional e de repetição;
- ✓ Conhecer os tipos de estruturas de dados homogêneas: (vetores e matrizes) e os dados heterogêneas (registros e arquivos) e praticá-los;
- ✓ Compreender a definição de arquivos, e conseguir criar arquivos sequenciais e diretos;
- ✓ Compreender os conceitos de programação estruturada, modularização, procedimento e função;

## VIII. CONTEÚDO PROGRAMÁTICO

### 1. Introdução à lógica

- 1.1 Conceituação
- 1.2 Histórico
- 1.3 Objetivos da lógica de programação
- 1.4 Sequência lógica,
- 1.5 Instruções,
- 1.6 Algoritmos,
- 1.7 Formas de representação de algoritmos
  - 1.7.1 Descrição Narrativa
  - 1.7.2 Fluxogramas
  - 1.7.3 Pseudo-código

### 2. Estruturas básicas de programação:

- 2.1 Tipos de dados,
- 2.2 Numéricos inteiros,
- 2.3 Numéricos reais,
- 2.4 Dados literais,
- 2.5 Dados lógicos

### 3. Tipos de expressões:

- 3.1 Instruções primitivas
  - 3.1.1 Atribuição,
  - 3.1.2 Saída de dados,
  - 3.1.3 Entrada de dados

### 4. Estruturas de controle condicional e repetição:

- 4.1 Comandos compostos,
- 4.2. Estrutura sequencial,
- 4.3. Estruturas de decisão
  - 4.3.1 Tipo "Se"
  - 4.3.2 Tipo "Escolha"
- 4.4 Laços contados
  - 4.4.1 Estrutura "Para-Faça"
- 4.5. Laços condicionais
  - 4.5.1 Estrutura "Enquanto-Faça"
  - 4.5.2 Estrutura "Repita-Até"
- 4.6 Aninhamentos/identação

### 5. Estruturas de dados

- 5.1 Estruturas Homogêneas:
  - 5.1.1 Vetores
  - 5.1.2 Matrizes
- 5.2 Estruturas heterogêneas
  - 5.2.1 Registros

### 6. Conceitos de programação modular:

- 6.1 Variáveis globais
- 6.2 Variáveis locais
- 6.3 Subalgoritmos
  - 6.3.1 Funções
  - 6.3.2 Procedimentos
- 6.4 Mecanismos de passagem de parâmetros.

## IX. METODOLOGIA DE ENSINO / DESENVOLVIMENTO DO PROGRAMA

- ✓ Aulas expositivas, dialogadas e práticas.

- √ Desenvolvimento de atividades pelos alunos (individuais e em equipes) orientadas pelo professor, abordando cada tema em estudo. Estas atividades além de dar um enfoque bastante prático à disciplina têm como objetivo mostrar ao aluno a importância do assunto em questão mediante a colocação deste no contexto computacional;
- √ Através do desenvolvimento dessas atividades e de atividades de pesquisa (dependendo do assunto em questão) pretende-se fazer com que o aluno seja constantemente avaliado mediante sua atuação dentro de cada tema que está sendo trabalhado.

## X. METODOLOGIA E INSTRUMENTOS DE AVALIAÇÃO

- A verificação do rendimento escolar compreenderá **frequência e aproveitamento** nos estudos, os quais deverão ser atingidos conjuntamente. Será obrigatória a frequência às atividades correspondentes a cada disciplina, ficando nela reprovado o aluno que não comparecer, no mínimo a 75% das mesmas.
- A nota mínima para aprovação na disciplina será 6,0 (seis). ( Art. 69 e 72 da Res. nº 17/CUn/1997).
- O aluno com frequência suficiente (FS) e média das notas de avaliações do semestre entre 3,0 e 5,5 terá direito a uma nova avaliação no final do semestre (REC), exceto as atividades constantes no art.70,§ 2º. A nota será calculada por meio da média aritmética entre a média das notas das avaliações parciais (MF) e a nota obtida na nova avaliação (REC). (Art. 70 e 71 da Res. nº 17/CUn/1997).

$$NF = \frac{MF + REC}{2}$$

- Ao aluno que não comparecer às avaliações ou não apresentar trabalhos no prazo estabelecido será atribuída nota 0 (zero). (Art. 70, § 4º da Res. nº 17/CUn/1997)

### • Avaliações

Primeira avaliação (P1) prova escrita e individual

Segunda avaliação (P2) prova escrita e individual

Os trabalhos, atividades e listas de exercícios, desenvolvidos em classe ou on-line (postados no AVA) compõem uma média denominada MT

$$MF = \left[ \left( \frac{P1 + P2}{2} \right) * 0,8 \right] + (MT * 0,2)$$

\* As provas poderão conter questões objetivas, objetivas mistas e dissertativas.

### Observações:

#### Avaliação de recuperação

- Não há avaliação de recuperação nas disciplinas de **caráter prático** que envolve atividades de laboratório (Res.17/CUn/97).

#### Nova avaliação

- Para pedido de segunda avaliação somente em casos em que o aluno, por motivo de força maior e plenamente justificado, deixar de realizar avaliações previstas no plano de ensino, deverá formalizar pedido de avaliação à Direção do Campus Araranguá dentro do prazo de 3 dias úteis apresentando comprovação. (Ver formulário)

## XI. CRONOGRAMA TEÓRICO/PRÁTICO

AULA (semana)	DATA	ASSUNTO
1	11/08/14 a 15/08/14	Apresentação Professor x Alunos; Apresentação do plano de ensino e Introdução à lógica de programação.
2	18/08/14 a 22/08/14	Introdução à lógica
3	25/08/14 a 29/08/14	2. Estruturas básicas de programação:
4	01/09/14 a 05/09/14	3. Tipos de expressões:
5	08/09/14 a 12/09/14	3. Tipos de expressões:
6	15/09/14 a 19/09/14	3. Tipos de expressões e 4. Estruturas de controle condicional e repetição:
7	22/09/14 a 26/09/14	4. Estruturas de controle condicional e repetição:
8	29/09/14 a 03/10/14	4. Estruturas de controle condicional e repetição e <b>Primeira avaliação:</b>

9	06/10/14 a 10/10/14	4. Estruturas de controle condicional e repetição:
10	13/10/14 a 17/10/14	4. Estruturas de controle condicional e repetição:
11	20/10/14 a 24/10/14	5. Estruturas de dados
12	27/10/14 a 31/10/14	5. Estruturas de dados
13	03/11/14 a 07/11/14	5. Estruturas de dados e <b>Segunda Avaliação.</b>
14	10/11/14 a 14/11/14	6. Conceitos de programação modular:
15	17/11/14 a 21/11/14	6. Conceitos de programação modular:
16	24/11/14 a 28/11/14	6. Conceitos de programação modular e <b>Terceira Avaliação.</b>
17	01/12/14 a 05/12/14	<b>Prova de reposição, nova avaliação (recuperação)</b>
18	08/12/14 a 12/12/14	<b>Publicação das notas</b>

**Obs 1:** O calendário está sujeito a pequenos ajustes de acordo com as necessidades das atividades desenvolvidas.  
**Obs 2:** O material disponibilizado no ambiente virtual de aprendizagem incluirá conteúdos preparatórios para os (ou complementares aos) encontros presenciais.  
**Obs 3:** Atendimento aos alunos deve ser agendado com o professor.

## XII. Feriados previstos para o semestre 2014.2:

DATA	
07/09/2014	Independência do Brasil
12/10/2014	Nossa Senhora Aparecida
02/11/2014	Finados
15/11/2014	Proclamação da República
25/12/2014	Natal

## XIII. BIBLIOGRAFIA BÁSICA

XAVIER, Gley Fabiano Cardoso. **Lógica de Programação**. 11. ed. São Paulo, SP: Ed. Senac, 2007.  
PESSOA, Marília; KRITZ, Sonia; PAIVA, Leonardo. **Construção de Algoritmos**. Rio de Janeiro, RJ: Ed. Senac Nacional, 2002.  
OLIVEIRA, Jayr F. de, MANZANO, José A. N. G., **ALGORITMOS: LÓGICA PARA DESENVOLVIMENTO DE PROGRAMAÇÃO DE COMPUTADORES**. 16a Ed., 2004.

## XIV. BIBLIOGRAFIA COMPLEMENTAR

CORMEN, Thomas H. **ALGORITMOS: TEORIA E PRÁTICA**. 1ª Ed., 2002.  
FORBELLONE, André Luiz Villar; EBERSPACHER, Henri Frederico. **Lógica de Programação: A Construção de Algoritmos e Estrutura de Dados**. 2ª. ed. São Paulo, SP: Makron Books, 2000.  
GUIMARÃES, A. M. **ALGORITMOS E ESTRUTURAS DE DADOS**. 1ª Ed., 1994.  
LOPES, A & Garcia, G. **INTRODUÇÃO À PROGRAMAÇÃO: 500 ALGORITMOS**. 1ª Ed., 2002.  
SEBESTA, Robert W. **CONCEITOS DE LINGUAGENS DE PROGRAMAÇÃO**. 5ª Ed., 2003.

Os livros acima citados constam na Biblioteca Universitária e Setorial de Araranguá.  
Algumas bibliografias também podem ser encontradas na Biblioteca Virtual da UFSC.

Prof. Jim Lau

Aprovado na Reunião do Colegiado do Curso

*B. U. 2014*

Prof. Dr. Eugênio Simão  
Coordenador do Curso de Graduação  
em Engenharia da Computação  
STAPE: 392745 Portaria nº 1071  
Coordenador do Curso