



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS ARARANGUÁ - ARA

PLANO DE ENSINO

SEMESTRE 2012.2

I. IDENTIFICAÇÃO DA DISCIPLINA:

| CÓDIGO  | NOME DA DISCIPLINA            | Nº DE HORAS-AULA SEMANAIS |          | TOTAL DE HORAS-AULA SEMESTRAIS |
|---------|-------------------------------|---------------------------|----------|--------------------------------|
|         |                               | TEÓRICAS                  | PRÁTICAS |                                |
| ARA7140 | Programação em Computadores I | -                         | 4        | 72                             |

| HORÁRIO         |   | MODALIDADE |
|-----------------|---|------------|
| TURMAS TEÓRICAS | TURMAS PRÁTICAS   | Presencial |
|                 | 01652A – 5-1830-4<br>01652B – 3-1830-2<br>01652B – 4-1830-2<br>02653 – 7-0820-2<br>01655A – 4-1420-2<br>01655A – 6-1420-2<br>01655B – 3-1830-2<br>01655B – 4-1830-2 |            |

II. PROFESSOR(ES) MINISTRANTE(S)

Prof. Carlos André de Sousa Rocha  
Email: [carlos.rocha@ararangua.ufsc.br](mailto:carlos.rocha@ararangua.ufsc.br)  
Prof. Marcel Campos Inocencio  
Email: [marcel.inocencio@ararangua.ufsc.br](mailto:marcel.inocencio@ararangua.ufsc.br)

III. PRÉ-REQUISITO(S)

| CÓDIGO | NOME DA DISCIPLINA                        |
|--------|---|
| -      | Esta disciplina não possui pré-requisitos |

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

Bacharelado em Tecnologias da Informação e Comunicação  
Graduação em Engenharia de Computação  
Graduação em Engenharia de Energia

V. JUSTIFICATIVA

Esta disciplina de caráter técnico prepara o discente para o desenvolvimento de soluções computacionais usando técnicas de programação. Além dos aspectos fundamentais da lógica de programação esta disciplina também foca o uso de uma primeira linguagem de programação.

VI. EMENTA

Conceito de algoritmo. Pseudo-código e fluxograma. Estrutura de um algoritmo. identificadores, palavras reservadas, variáveis, constantes, declaração de variáveis, comandos de entrada e saída, estruturas de controle de fluxo, estruturas de dados homogêneas (vetores e matrizes) e heterogêneas (registros). Tipos definidos pelo usuário. Modularização. Introdução à uma Linguagem de Programação de alto nível estruturada. Atividades em laboratório com a linguagem selecionada.

VII. OBJETIVOS

Objetivo Geral:

Tornar os alunos capazes de visualizar soluções computacionais para problemas através da aplicação dos conceitos da lógica de programação e dotá-los da capacidade de construção de programas, em linguagem de alto nível estruturada, que implementem as soluções vislumbradas.

#### **Objetivos Específicos:**

- Desenvolver o raciocínio lógico e abstrato do aluno;
- Familiarizar o aluno com o modelo seqüencial de computação;
- Apresentar técnicas e linguagens para representação e construção de algoritmos simples;
- Apresentar conceitos básicos de linguagens de programação;
- Treinar o aluno no processo básico de desenvolvimento de software (concepção, edição, execução e teste de programas de computador);
- Capacitar o aluno no uso de uma linguagem de alto nível.

### **VIII. CONTEÚDO PROGRAMÁTICO**

Conteúdo Teórico seguido de Conteúdo Prático com desenvolvimento de problemas em computador:

#### UNIDADE 1: Introdução [4 horas-aula]

- Conceito
- Algoritmo
- Funcionalidade de um algoritmo
- Exemplos de algoritmos

#### UNIDADE 2: Conceituação de elementos básicos para construção de um algoritmo [4 horas-aula]

- Constante
- Variável
- Identificador
- Palavra- reservada
- Operadores aritméticos, de atribuição, relacionais e lógicos
- Tipos de dados primitivos
  - Lógico
  - Caractere
  - Inteiro
  - Real
- Conceito de lógica
- Método para construção de um algoritmo
- Simbologia utilizada para a construção de fluxogramas
- Construção de algoritmos utilizando fluxogramas
- Estrutura de um pseudocódigo
- Construção de algoritmos em pseudocódigo

#### UNIDADE 3: Estruturas de controle de fluxo [16 horas-aula]

- Estruturas de seleção
  - Seleção simples
  - se então
  - se então senão
  - Seleção composta
  - escolha caso
- Estruturas de repetição
  - enquanto faça
  - repita até
  - para faça

#### UNIDADE 4: Variáveis compostas [16 horas-aula]

- Vetores unidimensionais e multidimensionais
  - caracterização
  - declaração
  - indexação
- Registros
  - caracterização
  - declaração

- registros com vetores

#### UNIDADE 5: Linguagens de programação [32 horas-aula]

- Conceituação de Linguagem de Programação
- Classificação das linguagens de programação com relação à similaridade com a linguagem natural
  - linguagem de máquina
  - linguagem simbólica
  - linguagem de alto nível
- Atividades de programação com uma linguagem de programação estruturada
- Codificação, compilação e execução
- Desenvolvimento de programas
  - Criação e manipulação de variáveis
  - Controle de fluxo
  - Variáveis compostas
  - Cadeia de caracteres (strings)
  - Ponteiros
  - Modularização
    - Definição de módulos
    - Procedimentos
    - Funções
    - Parâmetros
      - Por referência
      - Por valor

### IX. METODOLOGIA DE ENSINO / DESENVOLVIMENTO DO PROGRAMA

1. Aulas expositivas intercaladas com discussões. Material de apoio postado no Moodle. Desenvolvimento de trabalhos e exercícios;
2. Atividades práticas no computador, utilizando o ambiente de desenvolvimento de algoritmos Visualg e a Linguagem de Programação C.

### 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, no mínimo a 75% das mesmas (Frequência Suficiente - FS), ficando nela reprovado o aluno que não comparecer a mais de 25% das atividades (Frequência Insuficiente - FI).
- Serão realizadas quatro avaliações, sendo:
  - P1:** Prova Escrita 1 referente a resolução de problemas utilizando a aplicação de algoritmos.
  - P2:** Prova Escrita 2 referente a resolução de problemas utilizando algoritmos e a linguagem de programação C.
  - TP:** Trabalho Prático
- A Média Final (MF) será calculada da seguinte forma:
 
$$MF = [(P1 + P2) / 2] * 0,7 + TP * 0,3$$
- A nota mínima para aprovação na disciplina será MF >= 6,0 (seis) e Frequência Suficiente (FS). (Art. 69 e 72 da Res. nº 17/CUn/1997).

O aluno com Frequência Suficiente (FS) e média das notas de avaliações do semestre MF 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 Final (NF) 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)

#### 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**

• Pedidos de segunda avaliação somente para casos em que o aluno, por motivo de força maior e plenamente justificado, deixar de realizar avaliações previstas no plano de ensino, e deverá ser formalizado via requerimento de avaliação à Secretaria Acadêmica do Campus Araranguá dentro do prazo de 3 dias úteis apresentando comprovação. (Ver formulário)

**XI. CRONOGRAMA PRÁTICO**

| AULA (semana)   | DATA                | ASSUNTO  |
|-----------------|---------------------|--|
| 1 <sup>a</sup>  | 03/09/12 a 08/09/12 |  |
| 2 <sup>a</sup>  | 10/09/12 a 15/09/12 | <b>UNIDADE 1:</b><br>Apresentação da disciplina.<br>Apresentação do plano de ensino.<br>Introdução ao algoritmo.<br>Estrutura de um algoritmo.<br>Pseudo-código e fluxograma.<br>Introdução ao ambiente de desenvolvimento de algoritmos Visualg.<br>Comandos de entrada e saída de dados. |
| 3 <sup>a</sup>  | 17/09/12 a 22/09/12 | <b>UNIDADE 2:</b><br>Operadores: aritméticos, lógicos e relacionais.<br>Declaração de variáveis.<br>Tipos de variáveis.<br>Operador de atribuição.<br>Resolução de exercícios.   |
| 4 <sup>a</sup>  | 24/09/12 a 29/09/12 | <b>UNIDADE 3:</b><br>Estrutura de seleção simples, composta e encadeada, e exercícios.   |
| 5 <sup>a</sup>  | 01/10/12 a 06/10/12 | Estrutura de seleção de múltipla escolha e exercícios.   |
| 6 <sup>a</sup>  | 08/10/12 a 13/10/12 | Primeira avaliação – Prova Escrita 1: até estruturas de seleção de múltipla escolha.   |
| 7 <sup>a</sup>  | 15/10/12 a 20/10/12 | Estruturas de repetição <i>enquanto &lt;expr&gt; faça &lt;cmd&gt; e repita &lt;cmd&gt; até &lt;expr&gt;</i> , e exercícios.<br>Publicação do enunciado do Trabalho Prático.  |
| 8 <sup>a</sup>  | 22/10/12 a 27/10/12 | <b>UNIDADE 4:</b> Variáveis compostas homogêneas – vetores.  |
| 9 <sup>a</sup>  | 29/10/12 a 03/11/12 | Resolução de exercícios com vetores.   |
| 10 <sup>a</sup> | 05/11/12 a 10/11/12 | Variáveis compostas homogêneas – Matrizes.   |
| 11 <sup>a</sup> | 12/11/12 a 17/11/12 | Resolução de exercícios com matrizes.  |
| 12 <sup>a</sup> | 19/11/12 a 24/11/12 | <b>UNIDADE 5:</b><br>Introdução a linguagem de programação C<br>Ambiente de desenvolvimento integrado DevC++;<br>Tipos e declaração de variáveis<br>Comandos de entrada e saída de dados<br>Comandos de seleção em C: <i>if ... else; switch ... case</i> .                                |
| 13 <sup>a</sup> | 26/11/12 a 01/12/12 | Resolução de exercícios que envolvam comandos de seleção em C.<br>Comandos de repetição em C: <i>while; do while; for</i> .<br>Variáveis compostas em C: vetores e matrizes  |
| 14 <sup>a</sup> | 03/12/12 a 08/12/12 | Resolução de exercícios que envolvam comandos de repetição em C.<br>Conceitos de modularização – funções e procedimentos   |
| 15 <sup>a</sup> | 10/12/12 a 15/12/12 | Segunda avaliação – Prova Escrita 2: Estruturas de seleção e repetição e vetores usando a linguagem C.   |
| 16 <sup>a</sup> | 17/12/12 a 22/12/12 | Apresentação do Trabalho Prático (em laboratório).   |
| 17 <sup>a</sup> | 18/02/13 a 23/02/13 | Apresentação do Trabalho Prático (em laboratório).<br>Prova substitutiva.  |
| 18 <sup>a</sup> | 25/02/13 a 28/02/13 | Nova Avaliação (Prova de recuperação): Prova Prática em C compreendendo todo o conteúdo da disciplina.<br>Publicação de Notas.   |

\*As aulas referentes a semana do dia 03/09 a 08/09 serão repostas na forma de atividades extraclasse e/ou aulas extras.

**XII. Feriados previstos para o semestre 2012.2:**

| DATA |
|------|
|------|

|            |   |
|------------|---|
| 07/09/2012 | Independência do Brasil – Feriado Nacional(Lei nº 662/49    |
| 12/10/2012 | Nossa Senhora Aparecida – Feriado Nacional (lei nº 6802/80) |
| 02/11/2012 | Finados – Dia Santificado                                   |
| 15/11/2012 | Proclamação da República – Feriado Nacional (Lei nº 662/49) |

### XIII. BIBLIOGRAFIA BÁSICA

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene A. V. de. **Fundamentos da Programação de Computadores – Algoritmos, Pascal, C/C++ e Java**. 2. ed. Pearson, 2008.

FORBELLONE, André L. V.; EBERSPACHER, Henri F. **Lógica de Programação**. 3. ed. Pearson, 2005.

MIZRAHI, Victorine Viviane. **Treinamento em Linguagem C**. 2. ed. Pearson, 2008.

### XIV. BIBLIOGRAFIA COMPLEMENTAR:

SCHILD, Herbert. **C Completo e Total**. 3. ed. São Paulo: Pearson, 2006.

FARRER, H. et al. **Algoritmos Estruturados**. 3. ed. Rio de Janeiro: LTC, 2008.

GUIMARÃES, A. M.; LAGES, N. A. **Algoritmos e Estruturas de Dados**, Editora LTC, 1994.


ARAÚJO, Everton Coimbra de. **Algoritmos Fundamentos e Prática**. Visual Books, 2007.

MEDINA, Marco; FERTIG, Cristina. **Algoritmos e Programação – Teoria e Prática**. 2. ed.. São Paulo: Novatec, 2006.

Os livros acima citados constam na Biblioteca Universitária e Setorial de Araranguá. Algumas bibliografias também podem ser encontradas no acervo da disciplina, via sistema Moodle.

  
Prof. Carlos André de Sousa Rodha

Aprovado na Reunião do Colegiado do Curso 10/09/12

  
Coordenador do Curso  
Prof. Dra. Luciana Bolan Frigo  
Sub Coordenadora do Curso de Graduação  
em Engenharia da Computação  
SIAPE: 1805632 Portaria nº 107