



## PROGRAMA DE DISCIPLINA

Disciplina: <b>Introdução à Programação</b>	Código da Disciplina: <b>CCMP0041</b>
Carga horária total: <b>60</b>	Período de oferta: <b>2011.2</b>
Carga horária teórica: <b>15</b>	Turma: <b>CB</b>
Carga horária prática: <b>45</b>	
Professor responsável: <b>Marcus Vinícius Midená Ramos</b>	

### EMENTA:

Conceitos básicos de linguagens de programação. Histórico, classificação e principais aplicações de linguagens de programação. Modelos de execução de programas. Ferramentas de desenvolvimento. Nomes, valores e endereços. Tipos de dados. Constantes e variáveis. Expressões. Comandos de estruturação do fluxo de controle. Modularização. Escopo de nomes e tempo de vida de variáveis. Passagem de parâmetros. Recursividade. Tipos de dados definidos pelo usuário. Entrada e saída de dados. Arquivos. Uso em laboratório de uma linguagem de programação de alto-nível.

### OBJETIVO GERAL:

Tornar os estudantes capazes de visualizar soluções computacionais para problemas através da construção de programas, em linguagem de alto nível estruturada.

### OBJETIVOS ESPECÍFICOS:

- Familiarizar o estudante com o modelo seqüencial de computação;
- Apresentar conceitos básicos de linguagens de programação;
- Capacitar o estudante no uso da linguagem C;
- Treinar o aluno no processo básico de desenvolvimento de software (concepção, edição, execução e teste de programas de computador).

### CONTEÚDO PROGRAMÁTICO

- 1 Introdução**
  - 1.1 Conceito
  - 1.2 Computador



1.3 Computação

1.4 Programa

1.5 Algoritmo

1.6 Linguagem

## **2 Linguagens de Programação**

2.1 Conceitos básicos de linguagens

2.1.1 legibilidade

2.1.2 desempenho

2.1.3 segurança

2.1.4 portabilidade

2.1.5 confiabilidade

2.1.6 definição e uso de abstrações

2.1.7 expressividade

2.1.8 extensibilidade

2.1.9 flexibilidade

2.1.10 custos (treinamento, desenvolvimento, execução, manutenção etc)

2.1.11 linguagem de máquina x simbólica

2.1.12 linguagem de baixo nível x alto nível

2.2 Paradigmas de linguagens de programação

2.2.1 Imperativo

2.2.2 Orientado a objetos

2.2.3 Funcional

2.2.4 Lógico

2.3 Especificação de linguagens de programação

2.3.1 Métodos formais x informais

2.3.2 Sintaxe e BNF

2.3.3 Semântica

## **3 Modelos de execução de linguagens**

3.1 Interpretação

3.2 Compilação

3.3 Métodos híbridos

3.4 Máquina virtual x máquina real

## **4 Ambientes de programação**

4.1 Editores

4.2 Compiladores

4.3 Ambiente integrado de desenvolvimento (IDE)

4.4 Geradores de código

4.5 Programas auxiliares

## **5 Linguagem de Programação C**

5.1 Breve histórico

5.2 Características básicas

5.3 C padrão ANSI

5.4 Estrutura de um programa em C

5.5 Constantes

5.6 Palavras reservadas



- 5.7 Tipos primitivos
- 5.8 Declaração de variáveis
- 5.9 Operadores
  - 5.9.1 aritméticos
  - 5.9.2 atribuição
  - 5.9.3 relacionais
  - 5.9.4 lógicos
- 5.10 Funções de entrada e saída formatada
  - 5.10.1 printf
  - 5.10.2 scanf
- 5.11 Estruturas de controle de fluxo
  - 5.11.1 if, if else
  - 5.11.2 while
  - 5.11.3 do while
  - 5.11.4 for
  - 5.11.5 comando continue
  - 5.11.6 comando break
  - 5.11.7 switch case
- 5.12 Agregados homogêneos (Vetores unidimensionais e multidimensionais)
  - 5.12.1 caracterização
  - 5.12.2 declaração
  - 5.12.3 inicialização na declaração
  - 5.12.4 indexação
- 5.13 Agregados heterogêneos (registros)
  - 5.13.1 caracterização
  - 5.13.2 declaração
  - 5.13.3 inicialização na declaração
  - 5.13.4 seleção de campo
- 5.14 String
  - 5.14.1 caracterização
  - 5.14.2 declaração
  - 5.14.3 inicialização
  - 5.14.4 manipulação de funções para manipulação de strings
- 5.15 Funções
  - 5.15.1 caracterização
  - 5.15.2 forma geral
  - 5.15.3 comando return
  - 5.15.4 função main
  - 5.15.5 tipo void
  - 5.15.6 passagem de parâmetros por valor
  - 5.15.7 passagem de parâmetros por endereço
  - 5.15.8 escopo de nomes e tempo de vida de variáveis locais
  - 5.15.9 parâmetros formais
  - 5.15.10 variáveis globais
  - 5.15.11 protótipo de função
- 5.16 Tipos de dados definidos pelo usuário



- 5.16.1 definição de tipo
- 5.17 Ponteiros
  - 5.17.1 Ponteiros e agregados heterogêneos
  - 5.17.2 Ponteiros e agregados homogêneos
  - 5.17.3 Alocação dinâmica de memória
- 5.18 Mapa de memória
  - 5.18.1 Heap
  - 5.18.2 malloc e free
- 5.19 Recursão
- 5.20 Arquivos
  - 5.20.1 Arquivos texto
  - 5.20.2 Arquivos binários
- 6 Projeto de sistemas**
  - 6.1 Projeto de sistemas em linguagem C
  - 6.2 Definição de macros
  - 6.3 Diretivas para compilação condicional
  - 6.4 Arquivos de cabeçalho
  - 6.5 Divisão do projeto em arquivos separados
  - 6.6 Definição de bibliotecas de ligação estática
  - 6.7 Compilação separada com o uso do programa make

#### **METODOLOGIA:**

A disciplina será desenvolvida com aulas dialógico-expositivas e estudos dirigidos.

#### **RECURSOS MATERIAIS UTILIZADOS:**

Textos, quadro branco, pincel, notebook e data-show quando necessário.

#### **AVALIAÇÃO:**

A avaliação será distribuída em 02 notas (EE1 e EE2), valendo 10 pontos cada uma. A nota final será igual a  $(EE1 + EE2)/2$ .

#### **BIBLIOGRAFIA BÁSICA:**

1. SCHILDT, H. C **Completo e Total**. Pearson Education, 2006.
2. SEBESTA, R. W. **Conceitos de Linguagens de Programação**. 5ª ed. Bookman, 2006.

#### **BIBLIOGRAFIA COMPLEMENTAR:**

1. KERNIGHAN, B. W.; RITCHIE, D. M. **A Linguagem de Programação C ANSI**. Elsevier, 1989.
2. VAREJÃO, F. **Linguagens de Programação Java, C, C++ e outras**. Elsevier, 2004.



3. MOKARZEL, F.; SOMA, N. **Introdução à Ciência da Computação**. Elsevier, 2008.

ASSINATURA: _____	
-------------------	--

Carimbo do DRCA:	Emitido pelo DRCA em ____/____/____
------------------	-------------------------------------