



## PLANO DE ENSINO

<b>CURSO</b>	Engenharia Eletrônica	<b>MATRIZ</b>	544
--------------	-----------------------	---------------	-----

<b>FUNDAMENTAÇÃO LEGAL</b>	Resolução nº 63/06 – COEPP, de 17/11/06.
----------------------------	--

DISCIPLINA/UNIDADE CURRICULAR	CÓDIGO	PERÍODO	CARGA HORÁRIA (horas)		
			AT	AP	Total
Fundamentos de Programação 1	IF61C	1	51	51	102

AT: Atividades Teóricas, AP: Atividades Práticas.

<b>PRÉ-REQUISITO</b>	
<b>EQUIVALÊNCIA</b>	

### OBJETIVOS

Esta disciplina visa o aprendizado de um conjunto de habilidades e de conceitos fundamentais à prática de programação de computadores, bem como o desenvolvimento da abstração, da prática da solução conceitual e de estratégias para lidar com problemas complexos independentemente de paradigmas de codificação. Para isto, cobre conceitos básicos de programação de computadores, de resolução de problemas por algoritmos, de estruturas de dados básicas, assim como engloba o projeto e desenvolvimento de soluções de problemas por meio da construção de algoritmos e programas utilizando uma linguagem de programação.

### EMENTA

Computação e Sociedade; Conceitos Básicos em Computação; Introdução ao Paradigma Orientado a Objetos; Sintaxe e Semântica Básica de uma Linguagem de Programação de Alto Nível; Tipos de Dados Primitivos Básicos; Algoritmos e Resolução de Problemas; Atividades de laboratório.

### CONTEÚDO PROGRAMÁTICO

ITEM	EMENTA	CONTEÚDO
1	Computação e Sociedade	Impactos sociais e mudanças decorrentes da computação/informática na sociedade.
2	Conceitos Básicos em Computação	Histórico da computação. Conceitos básicos de arquitetura de computadores. Hardware e software. Bases numéricas. Histórico das linguagens de programação. Compilação x interpretação. Paradigmas e técnicas de programação.
3	Tipos de Dados Primitivos Básicos	Tipos de dados e abstração. Variáveis, tipos, expressões e atribuições. Precisão e erros de arredondamento.
4	Algoritmos e resolução de problemas	Conceitos básicos de algoritmos e estruturas de dados. Aplicações numéricas e não numéricas. Estratégias de resolução de problemas e de implementação de algoritmos. Estruturação de problemas.
5	Sintaxe e Semântica Básica de uma Linguagem de Programação de Alto Nível.	Conceitos básicos e histórico da linguagem considerada. Entrada e saída padrão. Operadores lógicos, aritméticos e binários. Estruturas de seleção. Estruturas de repetição. Vetores, matrizes e cadeias de caracteres. Arquivos. Conceito de escopo de identificadores. Conceito de função (método), parâmetros e retorno. Passagem de parâmetros. Recursão. Arquivos. Tópicos avançados.
6	Introdução ao Paradigma Orientado a Objetos	Histórico e apresentação de conceitos básicos: classes e objetos; métodos e atributos; construtores; encapsulamento. Introdução ao relacionamento entre objetos de classes: um para um, um para muitos, muitos para muitos.

PROFESSOR	TURMA
João Luiz Rebelatto (S11) / Edenilson José da Silva (S12)	S11 / S12

ANO/SEMESTRE	CARGA HORÁRIA (aulas)					
2015/01	AT	AP	APS	AD	APCC	Total
	51	45	06	00	00	102

AT: Atividades Teóricas, AP: Atividades Práticas, APS: Atividades Práticas Supervisionadas, AD: Atividades a Distância, APCC: Atividades Práticas como Componente Curricular.

DIAS DAS AULAS PRESENCIAIS						
Dia da semana	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
Número de aulas no semestre (ou ano)				51	45	

PROGRAMAÇÃO E CONTEÚDOS DAS AULAS (PREVISÃO)		
Dia/Mês ou Semana ou Período	Conteúdo das Aulas	Número de Aulas
19/Mar	Apresentação da disciplina.	3
20/Mar	Introdução a algoritmos. Resolução de problemas por meio da construção de algoritmos.	3
26/Mar	Exercícios <b>01</b> - fluxograma (apresentação dos alunos). Estruturas de decisão em algoritmos.	3
27/Mar	Estruturas de repetição em algoritmos. Tipos de dados e operadores.	3
09/Abr	Exercícios <b>02</b> (apresentação dos alunos). Especificação da Atividade Prática Supervisionada (APS). Tipos de linguagens de programação. Estrutura de um programa de computador. Variáveis, constantes e tipos de dados em C. Bases numéricas. Declaração vs definição de variáveis. Escopo de variáveis (variáveis locais e globais em C).	3
10/Abr	Procedimentos de entrada de dados via teclado e escrita na tela em C.	3
16/Abr	Exercícios <b>03</b> (apresentação dos alunos). Estruturas de decisão simples e compostas em Linguagem C.	3
17/Abr	Estruturas de repetição em Linguagem C.	3
23/Abr	Exercícios <b>04</b> (apresentação dos alunos). Vetores e matrizes em C ( <i>arrays</i> ).	3
24/Abr	<i>Arrays</i> de caracteres em C ( <i>strings</i> ).	3
30/Abr	Exercícios <b>05</b> (apresentação dos alunos). Definição de dados definidos pelo programador ( <i>structs</i> ). Definição de dados definidos pelo programador ( <i>union, enum</i> ).	3
07/Mai	Exercícios <b>06</b> (apresentação dos alunos). Aula de exercícios (Revisão).	3
08/Mai	<b>Avaliação 1:</b> Algoritmos e Introdução à Linguagem C (Vale 20%)	3
14/Mai	<b>Entrega da APS</b> (Vale 5,0%) Definição de funções e desenvolvimento estruturado de um programa. Declaração vs definição. Passagem de parâmetros por valor e referência. Recursividade.	3
15/Mai	Ponteiros. Operações com ponteiros. Ponteiros e <i>arrays</i> . Ponteiros e funções: passagem de ponteiros como argumentos e ponteiros como retorno de função.	3
21/Mai	Exercícios <b>07</b> (apresentação dos alunos)	3
22/Mai	Alocação dinâmica de memória	3
28/Mai	Exercícios <b>08</b> (apresentação dos alunos) Entrada e saída de arquivos texto.	3
29/Mai	Criação de projetos em arquivos separados (.h e .c)	3
11/Jun	Exercícios <b>09</b> (apresentação dos alunos) Introdução à Orientação a Objetos. Definição de classes e objetos. Modificadores de Acesso: <i>public, private</i> e <i>protected</i>	3
12/Jun	Métodos, atributos, construtores. Características principais da OO: Herança, Polimorfismo, Encapsulamento Diagramas de classe em UML.	3

<b>PROGRAMAÇÃO E CONTEÚDOS DAS AULAS (PREVISÃO)</b>		
<b>Dia/Mês ou Semana ou Período</b>	<b>Conteúdo das Aulas</b>	<b>Número de Aulas</b>
18/Jun	<b>Avaliação 2:</b> Programação Estruturada (Linguagem C) e Introdução à Orientação a Objetos (Vale 30%).	3
19/Jun	Especificação do projeto para os alunos. Interface Gráfica: Desenho de primitivas, de figuras e uso do mouse. Interface Gráfica: Detecção de obstáculos e animação 2D.	3
25/Jun	Desenvolvimento do Projeto.	3
26/Jun	Desenvolvimento do Projeto.	3
02/Jul	Desenvolvimento do Projeto.	3
03/Jul	Desenvolvimento do Projeto.	3
09/Jul	Desenvolvimento do Projeto.	3
10/Jul	Desenvolvimento do Projeto.	3
16/Jul	<b>Avaliação 3:</b> Entrega do relatório e Defesa do Projeto (Vale 30%).	3
17/Jul	Introdução à plataforma Arduino.	3
23/Jul	Recuperação de P1 e P2 e Encerramento da Disciplina.	3
APS	<u>Tópico:</u> Computação e sociedade. <u>Descrição:</u> Atividade individual. <u>Procedimentos:</u> Os alunos deverão ler material fornecido e responder questionário relacionado ao material. <u>Data da entrega/apresentação:</u> 14/05/2015. <u>Critérios de avaliação:</u> Qualidade das respostas e da apresentação.	6

### **PROCEDIMENTOS DE ENSINO**

#### **AULAS TEÓRICAS**

O tratamento das unidades temáticas nas aulas de teoria privilegiam as funções instrucionais de motivação e de orientação. Para tanto, empregam como formatos instrucionais a exposição, a discussão, o debate, a demonstração. Os materiais instrucionais adotados compreendem: quadro (branco ou de giz) projetor multimídia, arquivos .pdf, de texto e .html (i.e. apresentações relativas as unidades temáticas, textos técnicos, diagramas, etc.), navegadores internet, aplicativos de desenvolvimento de software.

#### **AULAS PRÁTICAS**

O tratamento das unidades temáticas nas aulas práticas privilegiam as funções instrucionais de aplicação, de avaliação e de controle. Para tanto, empregam como formatos instrucionais o trabalho em grupo (resolução de exercícios, discussão e implementação). Os materiais instrucionais adotados compreendem: quadro (branco ou de giz) projetor multimídia, arquivos .pdf, de texto e .html (i.e. apresentações relativas as unidades temáticas, textos técnicos, diagramas, etc.), navegadores internet, aplicativos de desenvolvimento de software.

#### **ATIVIDADE PRÁTICA SUPERVISIONADA**

Na APS, a aprendizagem será efetivada por meio de leitura e resolução de questionário.

### **PROCEDIMENTOS DE AVALIAÇÃO**

Aplicação de quatro avaliações e uma APS.

Duas Provas (P1 e P2), um Projeto (PROJ), uma APS e 9 listas de exercícios (EXE).

A média parcial será dada por:

$$MP = 0,2 * P1 + 0,3 * P2 + 0,3 * PROJ + 0,05 * APS + 0,15 * EXE.$$

Será aprovado o aluno com MP maior ou igual a 6,0 e frequência mínima de 75%.

A recuperação de P1 e P2 ocorrerá no final do semestre. Tendo nota máxima de 3,0 pontos (60% de P1 e P2)

O Projeto e a APS, por disporem de várias semanas para discussão com o professor e terem prazo de entrega, não terão recuperação.

### **REFERÊNCIAS**

#### **Referências Básicas:**

- FORBELLONE, A. L. V. & EBERSPÄCHER, H. F. Lógica de Programação : A construção de Algoritmos e Estruturas de Dados. Makron Books. 1993.
- FARRER, H. *et. alii*. Algoritmos Estruturados. LTC Editora. 1999.
- SCHILDT, H. C Completo e Total . Makron Books, 1997.
- DEITEL, H.M.; DEITEL, P.J. C++ Como Programar. 3ª Edição. Bookman. 2001

#### **Referências Complementares:**

- TANENBAUM A. S. Organização Estruturada de Computadores. 4a Ed. LTC Editora.
- GUIMARÃES, A. de M.; LAGES, N. A. de C. Algoritmos e Estruturas de Dados. LTC Editora. 1994.
- GERSTING, J. L. Fundamentos Matemáticos para a Ciência da Computação. LTC Editora. 2001.
- MIZRAHI, V. V. Treinamento em Linguagem C. Módulos 1 e 2. Makron Books. 1995.
- KERNIGHAN, B.; RITCHIE, D. C - A Linguagem de Programação. Ed. Campus. 1988.

Assinatura do Professor

Assinatura do Coordenador do Curso