



**UNIVERSIDADE FEDERAL DO ACRE**  
**PRÓ-REITORIA DE GRADUAÇÃO**  
**COORDENADORIA DE APOIO AO DESENVOLVIMENTO DO ENSINO**

**PLANO DE CURSO**

<b>Departamento:</b>	CCET - Centro de Ciências Exatas e Tecnológicas				
<b>Curso:</b>	Curso de Bacharelado em Sistemas de Informação				
<b>Disciplina:</b>	CCET005 - ALGORITMOS E LINGUAGEM DE PROGRAMAÇÃO				
<b>Código:</b>	CCET005	<b>Carga Horária:</b>	90 horas	<b>Créditos:</b>	4-1-0
<b>Pré-requisito:</b>	CCET186 – Introdução à Informática	<b>Semestre Letivo/Ano:</b>	2/2018		
<b>Professor:</b>	Jean Gonzaga Souza de Oliveira			<b>Titulação:</b>	Mestre

**1. Ementa:**

Introdução à lógica de programação. Conceitos fundamentais. Algoritmos. Elementos de um algoritmo. Formas de representação: português estruturado. Tipos de dados. Variáveis. Estrutura sequencial. Declaração de variáveis. Tipos de variáveis. Constantes. Comentários. Expressões, operadores e funções. Comandos básicos. Estruturas condicionais. Estruturas de repetição. Estruturas de dados homogêneas: vetores e matrizes. Programação estruturada em linguagem de alto nível. Funções e uso de ponteiros na linguagem de programação C.

**2. Objetivo Geral:**

Fornecer ao aluno condições de expressar soluções de forma algorítmica em uma linguagem de programação. Levar o aluno a entender os principais conceitos e usos de linguagens de programação. Permitir que o aluno desenvolva técnicas para construção e documentação de programa. Dar condições ao aluno de utilizar uma linguagem de programação para a resolução de problemas computacionais.

**3. Objetivos Específicos:**

Desenvolver a habilidade de construir modelos por meio da compreensão da atividade ou tarefa a ser modelada. Desenvolver o raciocínio lógico e abstrato. Familiarizar com o modelo sequencial de computação. Apresentar técnicas e pseudo-linguagens para construção e representação de algoritmos. Aprender técnicas para a confecção e desenvolvimento de algoritmos. Identificar problemas e fazer suposições e interpretações. Criar algoritmos para representar as interpretações feitas. Entender os principais conceitos no uso da linguagem de programação C. Apresentar os conceitos básicos da linguagem de programação C para construção de algoritmos. Fazer o processo de manutenção de algoritmos usando a linguagem C.

**4. Conteúdo Programático:**

<b>Unidades Temáticas</b>	<b>C/H</b>
<b>Unidade Temática 1</b> - Conceitos fundamentais. Linguagens de programação. Constantes. Variáveis. Formação das variáveis. Declaração de variáveis. Comentários.	10
<b>Unidade Temática 2</b> – Expressões aritméticas. Funções. Expressões lógicas. Relações. Operadores Lógicos. Comando de atribuição. Comandos de entrada e saída.	10
<b>Unidade Temática 3</b> - Estrutura sequencial. Estrutura condicional simples. Estrutura condicional composta. Estrutura de repetição. Estrutura de repetição com variável de controle.	10
<b>Unidade Temática 4</b> - Variáveis compostas homogêneas: Variáveis compostas unidimensionais e variáveis compostas bidimensionais	15
<b>Unidade Temática 5</b> - Modularização. Conceito de procedimento: caracterização, estrutura e parâmetros. Conceito de Função: caracterização,	15

estrutura e parâmetros. Passagem de parâmetros por valor. Passagem de parâmetros por referência. Recursão	
<b>Unidade Temática 6</b> - Noções de classificação e pesquisa de dados	10
<b>Unidade Temática 7</b> – Introdução à Linguagem C. História. Estilo de programação. Forma geral de um programa na linguagem C. Elementos da linguagem. Palavras reservadas. Declaração de variáveis e tipos de dados. Declaração de constantes. Expressões aritméticas e lógicas. Precedência entre operadores. Identificadores. Cadeia de caracteres. Números inteiros. Números em ponto flutuantes. Caracter. Strings. Operadores. Comando de atribuição. Funções pré-definidas. Comandos de entrada e saída. Comandos estruturados de seleção. Estruturas condicionais simples e compostas. Estruturas de repetição. Vetores e matrizes.	20
<b>5. Procedimentos Metodológicos:</b> A disciplina será trabalhada com aulas expositivas, onde serão fornecidos os componentes teóricos e será feita a prática de exercícios.	
<b>6. Recursos Didáticos:</b> Quadro branco, marcador, notebook e projetor multimídia.	
<b>7. Avaliação:</b> A avaliação será realizada mediante quatro provas escritas e dois trabalhos com a linguagem de programação C. As provas serão em datas especificadas previamente. A $N_1 = (P_1 + P_2) \div 2$ . A $N_2 = (P_3 + T_1) \div 2$ . O trabalho $T_1$ terá valor igual a 10,0.	
<b>8. Bibliografia:</b> CORMEN, Thomas H.; et. al. <b>Algoritmos: Teoria e Prática</b> . 2. Ed. Rio de Janeiro: Campus, 2002. 936 p. EVARISTO, Jaime. <b>Aprendendo a programar/programando em linguagem pascal</b> . 2. ed. São Paulo: Bookexpress, 2002. 244 p. FARRER, Harry. et. al. <b>Algoritmos estruturados</b> . 3. ed. Rio de Janeiro: Guanabara, 1999. 284 p. GUIMARÃES, Angelo de Moura; LAGES, Newton Alberto de Castilho. <b>Algoritmos e Estrutura de Dados</b> . Rio de Janeiro: LTC, 1994. 216 p. LOPES, Anita; GARCIA, Guto. <b>Introdução à Programação: 500 algoritmos resolvidos</b> . Rio de Janeiro: Campus, 2002. 488 p. WIRTH, Niklaus. <b>Algoritmos e estruturas de dados</b> . Rio de Janeiro: Prentice Hall, 1989. 255 p. ZIVIANI, Nívio. <b>Projeto de algoritmos: com implementações em Pascal e C</b> . 2. ed. São Paulo: Thomson Learning, 2004. 552 p.	
<b>Aprovação no Colegiado de Curso</b> (Regimento Geral da UFAC Art. 59, alíneas <b>b</b> e <b>n</b> ) <b>Data: 02 / 09 / 2017</b> Coordenador: Prof.: Olacir Rodrigues Castro Junior, Dr.	