



Universidade Estadual do Ceará
Centro de Ciências e Tecnologia
Curso de Bacharelado em Ciência da Computação

Disciplina	
Denominação: Programação Estruturada e Orientada a Objeto	Código: CT874
Número de Créditos: 06	Carga Horária: 104 horas
Pré-requisitos: Introdução a Computação (CT866)	
Ementa: Procedimentos e Funções. Ponteiros. Alocação dinâmica de memória. Estrutura de dados. Espaço de Nomes. Entrada e saída com arquivos. Técnicas para melhoria para rendimento em programação. Programação orientada a objetos. Classe e objetos. Encapsulamento de atributos e métodos. Abstração. Membros. Herança. Polimorfismo. Biblioteca padrão. Programação genérica (templates).	
Objetivos: Essa disciplina tem por objetivo aprofundar o conhecimento dos principais paradigmas de linguagens de programação.	
Conteúdo Programático:	
<ol style="list-style-type: none">1) Introdução<ol style="list-style-type: none">a) Revisão de vetores e matrizesb) String de caracteres2) Funções<ol style="list-style-type: none">a) Funçõesb) Funções sem tipo. O uso do void.c) Argumentos da função passados por valor e por referênciad) Valores padrões em argumentose) Número variável de parâmetrosf) Parâmetros da função principal (main)g) Sobrecarga de funçõesh) Funções inlinei) Recursividadej) Prototipação de funções3) Ponteiros<ol style="list-style-type: none">a) Operadores de endereço e de referênciab) Declaração de variáveis tipo ponteiroc) Ponteiros e arraysd) Inicialização de ponteirose) Aritmética de ponteirosf) Ponteiros para ponteirosg) Ponteiros para funções4) Alocação Dinâmica de Memória<ol style="list-style-type: none">a) Vazamento de memóriab) Alocação dinâmica5) Estrutura de Dados<ol style="list-style-type: none">a) Comando para estrutura de dados (struct)b) Ponteiros para estruturasc) Aninhamento de estruturasd) Tipos de dados definidos pelo usuárioe) Unions	

- f) Enumerations (enums)
- 6) Conceitos Avançados
 - a) Espaço de Nomes
 - b) Maquiagem de tipos (type casting)
- 7) Entrada e Saída com Arquivos
 - a) Acesso a disco (arquivos de texto para leitura e escrita)
 - b) Ponteiros para acesso a disco
- 8) Conceitos Fundamentais de Orientação a Objetos
 - a) Reutilização de código
 - b) Desenvolvimento e utilização de componentes
 - c) Comparação programação estruturada
- 9) Programação Orientada a Objetos
 - a) Abstração
 - b) Conceito de objeto
 - c) Polimorfismo
 - d) Classes
 - e) Herança e relacionamento entre classes
 - f) Sobrecarga de operadores
 - g) Construtor e destrutor de um objeto
 - h) Encapsulamento de atributos e métodos
 - i) Tratamento de exceções

Metodologia:

Aulas teórico-expositivas dos conteúdos

Uso de laboratório e aplicação prática

Desenvolvimento de um projeto

BIBLIOGRAFIA:**Básica:**

1. Schildt, Herbert. C - Completo e Total, 3ªEd. 1997.
2. Deitel, H. Java - Como Programar, 10ª Ed., 2016
3. Liang, Y. Daniel. Introduction to Java programming: brief version. Pearson, 2013.

Complementar:

4. Kazimoglu, Cagin, et al. "A serious game for developing computational thinking and learning introductory computer programming." *Procedia-Social and Behavioral Sciences* 47 (2012): 1991-1999.
5. Lutz, Mark. *Programming Python: Powerful Object-Oriented Programming*. " O'Reilly Media, Inc.", 2010.
6. Lutz, Mark. *Learning Python: Powerful Object-Oriented Programming*. " O'Reilly Media, Inc.", 2013.
7. Savitch, Walter J., et al. *Problem solving with C++*. Pearson Addison Wesley, 2012.
8. Harvey M. Deitel e Paul J. Deitel, *Java: How to program*, 4th. Ed. Prentice-Hall, 2001, ISBN 0-13-034151-7.