

## Plano de ensino

**Curso:** CCI-BAC - Bacharelado em Ciência da Computação

**Turma:** CCI122-03U - CIÊNCIA DA COMPUTAÇÃO FASE 03U

**Disciplina:** POO0001 - PROGRAMAÇÃO ORIENTADA A OBJETOS

**Período letivo:** 2018/1

**Carga horária:** 72

**Professor:** 211020727 - DIEGO BUCHINGER

### Ementa

- Conceitos de orientação a objetos. Decomposição de programas. Generalização e especialização. Agregação e composição. Herança e polimorfismo. Projeto orientado a objetos. Estudo de uma linguagem.

### Objetivo geral

- Oferecer meios para que o aluno entenda o funcionamento e conceitos básicos sobre programação orientada a objetos, projete um sistema orientado a objetos e conheça a linguagem de programação Java.

### Objetivo específico

- a) Dar subsídios aos acadêmicos para compreenderem o funcionamento e utilização dos conceitos associados a POO, como troca de mensagens, atributos, associações, agregações e composições;  
b) Apresentar aos acadêmicos os conceitos básicos sobre generalizações, interfaces, classes abstratas, polimorfismo e outros;  
c) Introduzir uma linguagem de programação orientada a objetos;  
d) Permitir que os acadêmicos apliquem os conceitos aprendidos na linguagem Java;  
e) Capacitar ao desenvolvimento de programas orientados a objeto.

### Conteúdo programático

1. Introdução
  - 1.1. Introdução ao Paradigma Orientado a Objetos
  - 1.2. Introdução a Linguagem de Programação Java
2. Conceitos Básicos de Orientação a Objetos
  - 2.1. Introdução a IDE Netbeans
  - 2.2. Classes, objetos, propriedade static, encapsulamento e método construtor
3. Introdução à Java Parte II
  - 3.1. Wrappers e conversão entre valores
  - 3.2. String, Math e Random
  - 3.3. Arrays - vetores e matrizes
  - 3.4. Manipulação de Arquivos
4. Conceitos Básicos de Orientação a Objetos II
  - 4.1. Enumeradores
  - 4.2. Associação, Agregação e Composição
  - 4.3. Destrutores e Garbage Collector
5. Conceitos de Orientação a Objetos I
  - 5.1. Herança
  - 5.2. Polimorfismo
6. Conceitos de Orientação a Objetos II
  - 6.1. Interface
  - 6.2. Classe Abstrata
7. Introdução à Java Parte III
  - 7.1. Tratamento de Exceções
8. Introdução à Java Parte IV
  - 8.1. Classe Arrays
  - 8.2. Coleções em Java: Collections
9. Introdução à Java Parte V
  - 9.1. Introdução à Interface Gráfica - Swing
  - 9.2. Componentes de interface gráfica - javax.swing
  - 9.3. Gerenciamento de layouts
  - 9.4. Tratamento de eventos
10. Padrões de Projeto
  - 10.1. Padrões criacionais
  - 10.2. Padrões estruturais
  - 10.3. Padrões comportamentais
  - 10.4. Padrão de projeto em Camadas
11. 12. Introdução à Java Parte V

## Plano de ensino

12.1. Multithreading - Thread e Runnable
12. 13. Conceitos de Orientação a Objetos III 13.1 Sobrecarga de Operadores (em C++)

### Metodologia

1. Aulas expositivas acompanhadas de práticas em laboratório, que objetivam a implementação dos conceitos apresentados nas aulas teóricas. Listas de exercícios para auxiliar na fixação do conteúdo apresentado. Trabalhos práticos relacionados aos conteúdos apresentados na aula e avaliar a capacidade do uso dos conceitos aprendidos. Provas teóricas para avaliar o conteúdo conceitual aprendido.

### Sistema de avaliação

1. Os alunos serão avaliados com base no seu desempenho nas provas, trabalhos e exercícios, sendo que ao final do semestre o aluno deverá ter comparecimento mínimo de 75% às aulas e desempenho mínimo de 70% nas avaliações. O grau de desempenho do aluno será avaliado com base no desenvolvimento das seguintes atividades e com os seguintes critérios:
  - a) Exercícios das aulas práticas (10% da média);
  - b) um trabalho de pesquisa (10% da média);
  - c) dois trabalhos de implementação (30% da média, 15% cada)
  - b) duas provas (50% da média, 25% cada).

### Bibliografia básica

1. DEITEL, H. M.; DEITEL, P. J. Java: Como Programar. 4 ed. Porto Alegre : Bookman, 2003.  
HORSTMANN, C. S.; CORNELL, G. Core Java 2. São Paulo : Makron Books, 2003.  
BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML Guia do Usuário. Rio de Janeiro : Campus, 2000.

### Bibliografia complementar

1. SANTOS, R. Introdução à Programação Orientada a Objetos usando Java. Rio de Janeiro : Elsevier, 2003.  
LARMAN, C. Utilizando UML e padrões. Porto Alegre : Bookman, 2002.  
METSKER, S. J. Padrões de Projeto em Java. Porto Alegre : Bookman, 2004.  
HAGGAR, Peter. Practical Java: Programming Language Guide. Reading, MA : Addison Wesley, 2000.  
ARNOLD, K., HOLMES, D. The Java programming language. 3 ed. Boston, MA : Addison Wesley, 2000.  
PAGE-JONES, M; PASCHOA, C. R. Fundamentos do desenho orientado a objeto com UML. São Paulo : Makron Books, 2001.