

**Instituto de Informática**  
**Departamento de Informática Aplicada**

## Dados de identificação

**Disciplina:** TÉCNICAS DE CONSTRUÇÃO DE PROGRAMAS

**Período Letivo:** 2016/2

**Período de Início de Validade:** 2016/2

**Professor Responsável pelo Plano de Ensino:** ERIKA FERNANDES COTA

**Sigla:** INF01120

**Créditos:** 4

**Carga Horária:** 60

## Súmula

Desenvolver e empregar técnicas que produzem programas de "boa qualidade": apresentação das técnicas utilizadas para este fim; construção, durante o semestre, de um sistema de médio porte, onde estas técnicas devem ser exercitadas.

## Currículos

Currículos	Etapas Aconselhadas	Natureza
ENGENHARIA DE COMPUTAÇÃO	4	Obrigatória
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO	4	Obrigatória
BIOTECNOLOGIA MOLECULAR		Eletiva
BIOINFORMÁTICA	5	Alternativa

## Objetivos

Capacitar o aluno a implementar, testar e refinar programas complexos a partir de um conjunto de técnicas de programação que visem a qualidade do software. Como parte deste objetivo, esta disciplina oferece os conceitos básicos de qualidade de software, modularidade, tipos abstratos de dados e uma introdução ao paradigma da programação orientada a objetos.

## Conteúdo Programático

<b>Semana:</b> 1
<b>Título:</b> Introdução e Motivação
<b>Conteúdo:</b> Apresentação e discussão do problema do software.
<b>Semana:</b> 1
<b>Título:</b> Qualidade de software: fundamentos, critérios e fatores de qualidade
<b>Conteúdo:</b> Conceitos de qualidade de software
<b>Semana:</b> 2
<b>Título:</b> Modularidade
<b>Conteúdo:</b> Critérios de modularidade. Acoplamento e coesão. Regras e princípios de modularidade.
<b>Semana:</b> 3
<b>Título:</b> Análise de código
<b>Conteúdo:</b> Análise de código fonte sob os critérios de qualidade interna discutidos em aula.
<b>Semana:</b> 4
<b>Título:</b> Projeto de software visando o reuso
<b>Conteúdo:</b> Dificuldades para o reuso no paradigma procedural. Técnicas de programação e requisitos de linguagens para o reuso de código.
<b>Semana:</b> 4
<b>Título:</b> Decomposição de software. Tipos abstratos de dados.
<b>Conteúdo:</b> Conceito matemático de tipos abstratos de dados.
<b>Semana:</b> 5 a 6
<b>Título:</b> Programação Orientada a Objetos I
<b>Conteúdo:</b> Introdução à programação orientada a objetos: classes e objetos.
<b>Semana:</b> 7 a 8
<b>Título:</b> Programação Orientada a Objetos II
<b>Conteúdo:</b> Introdução à programação orientada a objetos: herança e polimorfismo.
<b>Semana:</b> 9 a 10
<b>Título:</b> Convenções e boas práticas de programação

**Conteúdo:** Convenções e boas práticas de programação: identificadores, tipos de dados, controle.

**Semana:** 11 a 12

**Título:** Conceitos básicos de Teste de Software.

**Conteúdo:** Teste de desenvolvedor. Teste funcional, teste de classes.

**Semana:** 13

**Título:** Depuração de código

**Conteúdo:** Técnicas para depuração de código.

**Semana:** 14

**Título:** Refatoração

**Conteúdo:** Conceito e práticas de refatoração.

**Semana:** 15

**Título:** Avaliação

**Conteúdo:** Avaliações e apresentação de trabalhos.

## Metodologia

As aulas serão de cunho teórico-expositivo intercaladas com discussões sobre os exercícios e os trabalhos extra-classe, além de aulas em laboratório, quando couber. Além disso, estão previstas aulas de discussão onde os alunos irão identificar e discutir técnicas específicas de programação ou aspectos de linguagens de programação que afetem a qualidade do software. Por fim, um trabalho prático de programação será definido para a fixação das técnicas discutidas durante as aulas.

## Carga Horária

Teórica: 40

Prática: 20

## Experiências de Aprendizagem

Exercícios extra-classe de programação e análise de código.

Um trabalho prático de maior porte para exercitar os conceitos de abstração de dados, refatoração e teste.

## Critérios de avaliação

Serão realizadas até 2 (duas) provas em datas a serem fixadas. As provas envolvem o conteúdo de toda disciplina ministrado até a prova. Serão realizados trabalhos teóricos e/ou práticos para consolidação e aplicação dos conceitos vistos em aula.

Os trabalhos práticos são organizados em sua maioria como atividades extraclasse. A realização dos trabalhos práticos é caracterizada com atividade autônoma, mas estão previstas aulas de acompanhamento com o professor para resolver dúvidas da elaboração do trabalho. Os trabalhos práticos serão realizados em grupo e seus desenvolvimentos (parciais e/ou final) enviados via moodle nas datas indicadas.

A cada prova e a cada trabalho será atribuído um grau entre 0 (zero) a 10 (dez). Nas atividades em grupo será avaliada a contribuição individual de cada integrante.

A nota de participação inclui a avaliação sobre realização/participação de exercícios e dinâmicas propostos, aulas de laboratório, posicionamento quanto a conteúdo e dúvidas, qualidade de participação em aula e motivação durante o desenvolvimento dos trabalhos, e assiduidade do aluno.

O conceito final do aluno será atribuído levando-se em consideração a sua participação nas atividades em classe e extra-classe (10%), a média dos trabalhos realizados pelo aluno (40%) e a média das provas (50%).

A conversão da média numérica para conceitos é feita por meio da seguinte tabela:

9,0 = MG ≤ 10,0 : conceito A (aprovado).

7,5 = MG < 9,0 : conceito B (aprovado).

6,0 = MG < 7,5 : conceito C (aprovado).

0,0 = MG < 6,0 : conceito D (reprovado).

Alunos com frequência inferior a 75% das aulas são reprovados por FF.

## Atividades de Recuperação Previstas

O aluno que obtiver conceito final D pode realizar uma prova ou atividade (a critério do professor) de recuperação versando sobre todo o conteúdo da disciplina, com data a ser definida pelo professor. Se a nota obtida nessa prova for igual ou superior a 6,0, o conceito mudará para C.

Observações: Para poder realizar a prova de recuperação, o aluno deve ter realizado ao menos uma das provas/trabalhos teóricos E ter entregue o trabalho prático. Os que não se enquadrarem nesta situação permanecerão com conceito D.

A recuperação de cada prova individualmente será realizada somente para os casos previstos na legislação: saúde, parto, serviço militar, convocação judicial, luto, etc., devidamente comprovados, em data e horário a serem definidos pelo professor.

## Prazo para Divulgação dos Resultados das Avaliações

A avaliação de cada atividade ou prova será divulgada em até 3 semanas após realização da mesma ou 72 horas antes da prova/atividade de recuperação.

## Bibliografia

### Básica Essencial

McConnell, Steve; Tortello, João Eduardo Nóbrega. Code complete :um guia prático para a construção de software. Porto Alegre: Bookman, 2005. ISBN 8536305045.

Meyer, Bertrand. Object-oriented software construction. Upper Saddle River: Prentice Hall PTR, c1997. ISBN 0136291554.

### Básica

Kent Beck. Test Driven Development: By Example. Addison-Wesley Professional, 2002. ISBN 978-0321146533.

Robert C. Martin. Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall, 2008. ISBN 978-0132350884.

Robert C. Martin. Agile Software Development, Principles, Patterns, and Practices. Prentice Hall, 2002. ISBN 978-0135974445.

Ron Patton. Software Testing. USA: Sams Publishing, 2006. ISBN 0-672-32798-8.

### Complementar

Bertrand Meyer. Touch of Class - Learning to Program Well with Objects and Contracts. Springer-Verlag, 2009. ISBN 3540921443.

Horstmann, Cay S.. Object-oriented design. Hoboken: John Wiley, c2006. ISBN 0471744875.

McConnell, Steve. A practical handbook of software construction. Redmond: Microsoft Press, 2004. ISBN 9780735619678.

## Outras Referências

Título	Texto
Material disponível no Moodle do INF	A disciplina conta com uma entrada no Moodle do INF onde são colocados materiais complementares para consulta e estudo, especificação das atividades práticas desenvolvidas em aula e extra-classe, cronograma atualizado e cópia do plano de ensino.

## Observações

*Nenhuma observação incluída.*