



Universidade Federal do ABC (UFABC)  
Centro de Matemática, Computação e  
Cognição (CMCC)

*Bacharelado em Ciência da Computação*

## Plano de Ensino

*Disciplina: Programação Orientada a Objetos(BC1501)*

### Dados de Identificação

- **Ano Letivo:** 2020
- **Quadrimestre:** S
- **Sala:** Online - <https://meet.google.com/bnf-idqt-khd>
- **Turno:** Matutino
- **Campus:** Santo André
- **Créditos (T-P-I):** 2-2-4

### *Nota*

O escopo deste plano de ensino esta limitado ao **Quadrimestre Suplementar**, Ano 2020, criado em decorrência da Pandemia do **COVID-19** no Brasil e região. O plano busca seguir as diretrizes definidas na Resolução Nro 240/2020 - CONSEPE, de 15 de Julho de 2020, Publicada no Boletim de Serviço nro 963/2020. *Não é autorizada a redistribuição e/ou uso público de qualquer material disponibilizado na disciplina.*

### Sumário

1	Professor(es)	2
2	Ementa	2
3	Objetivo	2
4	Competências	3
5	Recomendações	3

<b>6</b>	<b>Horários</b>	<b>3</b>
<b>7</b>	<b>Cronograma / Mapa de Atividades</b>	<b>4</b>
<b>8</b>	<b>Estratégias Didáticas</b>	<b>6</b>
<b>9</b>	<b>Avaliação de Aprendizagem</b>	<b>6</b>
9.1	Composição do Conceito . . . . .	6
9.2	Análise de Desempenho e Conceitos . . . . .	6
9.3	Cálculo de Conceitos . . . . .	6
9.4	Atividades . . . . .	8
9.5	Reprovação . . . . .	8
9.6	Recuperação . . . . .	8
<b>10</b>	<b>Ferramentas</b>	<b>9</b>
<b>11</b>	<b>Atividades de Apoio</b>	<b>9</b>
11.1	Atendimento Extraclasse . . . . .	9
11.2	Canal de Interação . . . . .	10
<b>12</b>	<b>Informações adicionais</b>	<b>10</b>
<b>13</b>	<b>Bibliografia</b>	<b>11</b>
13.1	Básica . . . . .	11
13.2	Complementar . . . . .	11

## 1 Professor(es)

O seguinte professor sera responsável pela condução da disciplina “Programação Orientada a Objetos” no QS.2020.

### 1. Prof. Dr. Flávio Eduardo Aoki Horita:

- Turma A1
- E-mail: flavio.horita@ufabc.edu.br
- Website: <http://www.flaviohorita.com>

## 2 Ementa

Conceitos básicos: classes, objetos, mensagens, encapsulamento, herança, polimorfismo. Programação orientada a objetos utilizando uma linguagem de programação orientada a objetos (Java). Análise e projeto orientados a objetos. UML. Padrões de projeto de software.

## 3 Objetivo

Apresentar os conceitos básicos e técnicas da programação orientada a objetos.

## 4 Competências

Ao final da disciplina o aluno deverá ser **capaz** de:

- Compreender os conceitos básicos da programação orientada a objetos;
- Aplicar adequadamente os conceitos e técnicas de programação orientada a objetos no projeto e desenvolvimento de sistemas computacionais.

## 5 Recomendações

A disciplina apresentada abaixo compreende o **requisito mínimo** para um bom aproveitamento e aprendizagem do conteúdo ministrado nessa disciplina.

- Processamento da Informação
- Programação Estruturada

OBS: Caso o aluno não tenha cursado (e obtido aprovação) a disciplina recomendada, recomenda-se um esforço dobrado na carga horária dedicada as atividades individuais (I) dessa disciplina, bem como uma atenção redobrada nas aulas teóricas. Não serão aceitas, em qualquer momento dessa disciplina, justificativas quando ao não entendimento dos conceitos atribuídos para as recomendadas.

## 6 Horários

Tabela 1 apresenta a grade de horas da disciplina.

Tabela 1: Horários das Aulas

Horário	Segunda	Terça	Quarta	Quinta	Sexta
08:00-10:00		Video Aulas			
10:00-12:00					Aula
14:00-16:00					
16:00-18:00					
19:00-21:00					
21:00-23:00					

## 7 Cronograma / Mapa de Atividades

O calendário dos conteúdos Teóricos e Práticos, assim como o cronograma de Provas e Apresentações estão apresentados no mapa de atividades das Tabelas 2 e 3. A descrição das atividades no mapa de atividades.

Tabela 2: Cronograma 1/2

Semana	Horas (T+P)	Tema / Subtema	Objetivos específicos	Atividades teóricas	Atividades práticas
1 21/09/2020 a 27/09/2020	4h	Introdução ao Java <ul style="list-style-type: none"> <li>↳ Tipos de dados</li> <li>↳ Entrada/Saída</li> <li>↳ Operadores aritméticos, relacionais, lógicos</li> <li>↳ Estrutura condicional</li> <li>↳ Estruturas de repetição</li> <li>↳ Vetores e matrizes</li> <li>↳ Métodos</li> <li>↳ Estilo de codificação</li> </ul>	Aplicar a linguagem Java para implementação de programas, incluindo os programas propostos na disciplina.	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
2 28/09/2020 a 04/10/2020	4h	Classes, objetos, encapsulamento <ul style="list-style-type: none"> <li>↳ Classes, atributos e métodos</li> <li>↳ Classe vs Instância</li> <li>↳ Modificadores de acesso</li> <li>↳ Pacotes</li> <li>↳ UML (representação de uma classe)</li> <li>↳ UML (relacionamentos: associação, agregação e composição)</li> </ul>	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana. Entender o diagrama de classes (com os conceitos passados até esta semana)	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
3 05/10/2020 a 11/10/2020	4h	Construtores, sobrecargas <ul style="list-style-type: none"> <li>↳ Construtores</li> <li>↳ Sobrecargas</li> </ul>	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
4 12/10/2020 a 18/10/2020	4h	Herança <ul style="list-style-type: none"> <li>↳ Herança</li> <li>↳ Polimorfismo</li> <li>↳ UML (herança no diagrama de classes)</li> </ul>	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana. Entender o diagrama de classes (com os conceitos passados até esta semana)	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
5 19/10/2020 a 25/10/2020	4h	Interfaces <ul style="list-style-type: none"> <li>↳ Classe abstrata</li> <li>↳ Interfaces</li> <li>↳ UML (interface no diagrama de classes)</li> </ul>	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana. Entender o diagrama de classes (com os conceitos passados até esta semana)	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
6 26/10/2020 a 01/11/2020	4h	Construtores, cópia de objetos, final <ul style="list-style-type: none"> <li>↳ Chamada a outros construtores</li> <li>↳ Acesso a construtores</li> <li>↳ Cópia de objetos</li> <li>↳ Modificador <i>final</i></li> </ul>	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas) - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					

Tabela 3: Cronograma 2/2

7 02/11/2020 a 08/11/2020	4h	Estruturas lineares no contexto de orientação a objetos — Modificadores de acesso em classes  — Aplicação dos conceitos vistos em estruturas lineares	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
8 09/11/2020 a 15/11/2020	4h	Tratamento de exceções — Hierarquia de exceções  — Estruturas para lançar e capturar exceções	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
9 16/11/2020 a 22/11/2020	4h	Generics	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
10 23/11/2020 a 29/11/2020	4h	Padrões de projeto de software	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
11 30/11/2020 a 06/12/2020	4h	Android	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	Exercícios para submissão pelo Moodle.
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					
12 07/12/2020 a 12/12/2020	4h	Revisão	Entender os conceitos de programação orientada a objetos apresentados até esta semana. Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.	- Assistir videoaulas (gravadas)  - Ler material	
Feedback, comunicação e avaliação (exercícios de programação): avaliação dos exercícios.					

- **Horas:** a coluna horas na tabela é soma de T+P. Contudo, pode ser necessário tempo adicional para estudo, conforme indicado no T-P-I da disciplina.
- **Assistir videoaulas (gravadas):** serão disponibilizados links para videoaulas sobre o conteúdo da disciplina. É necessário um navegador web com capacidade de executar vídeos armazenados no Google drive (alguns vídeos podem estar no Youtube). Durante as videoaulas, além de apresentar o conteúdo, podem ser realizados exemplos práticos de implementação dos conceitos (sistema operacional Ubuntu Linux 18.04, Open Java Development Kit (OpenJDK 11)). Ao longo dos vídeos, poderão ser propostos exercícios (não avaliativos).
- **Ler material (PDF):** será disponibilizado material para leitura (por exemplo, slides da aula em PDF). É recomendável ler o material, além de assistir às videoaulas. Exercícios para submissão pelo Moodle: serão disponibilizados exercícios ao longo do quadrimestre. Os exercícios deverão ser submetidos pelo Moodle. Cada exercício pode ter pontuação/peso e prazo de entrega diferentes (essas informações estarão em cada exercício no Moodle). Exercícios de programação poderão ser avaliados por sistema de correção automática, mas a nota atribuída automaticamente poderá ser modificada pelo professor. Os exercícios de programação para nota possuem a indicação no Moodle.

## 8 Estratégias Didáticas

As estratégias didáticas adotadas compreendem os seguintes itens:

- Um conjunto de **video-aulas** liberadas via Ambiente Virtual de Aprendizagem;
- **Aula** síncrona utilizando Google meet para realizar exercícios e tirar dúvidas com os alunos;
- Listas de **exercícios** para reforço e avaliação do aprendizado.

A parte prática pode ser realizada em ambiente de desenvolvimento montado em um computador pessoal com Ubuntu Linux 18.04 e OpenJDK 11.

## 9 Avaliação de Aprendizagem

Esta seção do plano de ensino apresenta os elementos definidos para avaliação da aprendizagem do conteúdo ministrado na disciplina.

### 9.1 Composição do Conceito

A avaliação desta disciplina será composta pelos itens apresentados na Tabela 4.

Tabela 4: Composição do conceito

Item	Descrição	Peso para conceito final
E1 .. E <sub>n</sub>	Exercícios	P1...P <sub>n</sub>

### 9.2 Análise de Desempenho e Conceitos

A avaliação da disciplina será baseada exclusivamente em conceitos. Para cada item listado na seção anterior, será atribuído um conceito, conforme Tabela 5.

### 9.3 Cálculo de Conceitos

O cálculo detalhado nessa seção pode ser utilizado para definir o conceito de quaisquer itens de avaliação dessa disciplina (entende-se: provas, atividades em laboratório, projetos, atividades de recuperação, extraclasse, etc). O conceito será determinado pela **média aritmética ponderada** apresentada na Fórmula 1.

$$C_f = \text{ClassificaConceito}\left(\frac{p_1 \times x_1 + p_2 \times x_2 + \dots + p_n \times x_n}{p_1 + p_2 + \dots + p_n}\right) \quad (1)$$

Onde  $C_f$  é igual ao conceito final,  $p$  é o peso e  $x$  é o valor do item/critério. Além disso, a função  $\text{ClassificaConceito}(x)$  retorna o respectivo conceito de acordo com Tabela 6, sendo  $x$  limitado apenas 1 casa decimal, sem arredondamentos; enquanto a função  $\text{max}(x, y)$  retorna o maior valor entre  $x$  e  $y$ .

Tabela 5: Conceitos

Conceito	Descrição
A	Desempenho excepcional, demonstrando excelente compreensão da disciplina.
B	Bom desempenho, demonstrando boa capacidade de uso dos conceitos da disciplina.
C	Desempenho adequado, demonstrando capacidade de uso dos conceitos da disciplina e capacidade para seguir estudos mais avançados.
D	Aproveitamento mínimo dos conceitos da disciplina com familiaridade parcial do assunto, mas demonstrando deficiências que exigem trabalho adicional para prosseguir em estudos avançados.
F	Aproveitamento mínimo dos conceitos da disciplina com familiaridade parcial do assunto, mas demonstrando deficiências que exigem trabalho adicional para prosseguir em estudos avançados.
O	Reprovado por ausência de avaliação.

Tabela 6: Classificação do Conceito

Conceito	Faixa de valores
A	$x \geq 8,5$
B	$7,5 \leq x < 8,5$
C	$6,0 \leq x < 7,5$
D	$4,5 \leq x < 6,0$
F	$x < 4,5$

Para exemplificar a aplicação da Formula 9.3, consideramos os seguintes itens de avaliação com os seguintes valores e pesos (fictícios): Peso 10 e Valor 4 para L, Peso 5 e Valor 2 para A, e Peso 5 e valor 5 para P. O conceito final será atribuído como descrito na Formula 2.

$$\begin{aligned}
 C_f &= \text{ClassificaConceito}\left(\frac{5 \times 4 + 3 \times 2 + 2 \times 5}{5 + 3 + 2}\right) \\
 C_f &= \text{ClassificaConceito}\left(\frac{20 + 6 + 10}{10}\right) \\
 C_f &= \text{ClassificaConceito}\left(\frac{36}{10}\right) \\
 &\text{ClassificaConceito}(3,6) = F
 \end{aligned} \tag{2}$$

**IMPORTANTE:** Os valores serão limitado apenas a 1 primeira casa decimal, sem arredondamentos, ou seja, 8,99 é igual a 8,9 (Conceito B) e não 9,0 (Conceito A).

## 9.4 Exercícios

Quando houver, serão considerados como exercícios, aqueles cobrados para entrega durante o próprio horário de aula ou feitos em casa. As entregas serão on-line por meio de plataforma indicada pelo Professor.

Para o cálculo dos Exercícios, serão utilizados os critérios abaixo.

- Exercícios que valem até 1 ponto: são os exercícios mais simples, é necessário passar em todos os casos de teste para obter 1 ponto. Caso contrário, ficará com zero.
- Exercícios que valem mais de 1 ponto: exercícios mais complexos onde a nota dependerá dos casos de testes que ele passar, sejam estes casos fechados ou abertos. Dessa forma, estes exercícios podem receber pontuação parcial.
- Esses critérios são aplicados pelo corretor automático no Moodle. Entretanto, o professor poderá alterar a pontuação manualmente (por exemplo, devido a pontos que o corretor automático não pode identificar, problemas/erros na avaliação ou identificação de plágios).

## 9.5 Reprovação

O aluno será reprovado automaticamente (**sem direito ao mecanismo de recuperação**) nas seguintes hipóteses:

- **Plagiar** qualquer trabalho ou atividade.

O aluno será reprovado, mas **com direito ao mecanismo de recuperação** (ver Seção 9.6), nas seguintes hipóteses:

- **Obter** o conceitual final F ou D.

No caso de **ausência de alguma avaliação descrita na Tabela 4**, o valor atribuído será “0”.

## 9.6 Recuperação

O aluno terá o direito de fazer o uso de mecanismos de recuperação conforme a Resolução no 182 - 23/10/14. Para utilizar o mecanismo de recuperação, o aluno deve estar aprovado com o conceito D ou reprovado com o conceito F, desde que respeitadas as condições descritas no item 9.5.

O mecanismo de recuperação não poderá ser aplicado em período inferior a 72 horas após a divulgação dos conceitos das avaliações regulares. Este mecanismo quando aplicado poderá ser composto por **um ou mais** itens abaixo, ficando a critério do professor defini-lo:

- *Prova*: Substitui incondicionalmente a menor nota de prova regular, independente do seu peso (se houver mais de uma prova);
- *Atividades*: Envio de Atividades incompletas ou não entregues.



- *Trabalhos, Seminários, Relatórios, etc:* Poderão ser solicitados a entrega de outros elementos de avaliações previstos para a disciplina. Estas avaliações irão substituir os seus respectivos conceitos regulares, mas com desconto.

Sejam quais forem os mecanismos de recuperação utilizados, este(s) ira(ão) **elevantar ou manter o conceito final do aluno**, de acordo com o seu desempenho. O conceito final obtido na disciplina, após o mecanismo de recuperação, **não excederá um nível acima do conceito obtido antes da recuperação**. Por exemplo, se o aluno estiver com F, poderá ficar com F ou D, e se o aluno estiver com D, poderá ficar com D ou C. Para subir de conceito, o aluno deve se enquadrar nas seguintes situações.

**O aluno deverá solicitar com antecedência a intenção de realizar o mecanismo de recuperação, conforme prazo e canal estabelecido pelo Professor.** A ausência desta confirmação **impedirá** o aluno de realizar o mecanismo de avaliação. Por fim, os critérios de avaliação da recuperação serão informados quando da liberação dos conceitos.

## 10 Ferramentas

Para acompanhar a disciplina é necessário computador com acesso à internet:

- Recomenda-se o uso do sistema operacional Ubuntu Linux 18.04 64-bit;
- Caixa de som ou fone de ouvido;
- Navegador web compatível com o Moodle UFABC e Google Meet; O navegador web deve permitir a execução de vídeos a partir do Google drive.
- Editor de texto gedit (outros editores podem ser compatíveis também, como, por exemplo, Notepad++ e Sublime).
- Open Java Development Kit (OpenJDK) 11 ? Será usada a versão disponível no Ubuntu Linux 18.04 64-bit

## 11 Atividades de Apoio

Essa seção detalha algumas das atividades extra que visam apoiar os alunos participantes na disciplina quanto a condução de seu aprendizado.

### 11.1 Atendimento Extraclasse

Esta disciplina prevê um horário de atendimento extraclasse para atividades de apoio aos estudantes regulares desta turma, conforme disposto na Resolução CONSUNI 183, de 31 de outubro de 2017.

Os horários de atendimento semanal serão realizadas nos seguintes dias, locais e horários:

*Teoria*

- **Dia da semana:** Sexta-feira
- **Local:** <https://meet.google.com/bnf-idqt-khd>
- **Horário:** das 13:00 as 15:00
- **Professor:** Dr. Flávio Eduardo Aoki Horita ([flavio.horita@ufabc.edu.br](mailto:flavio.horita@ufabc.edu.br))

**IMPORTANTE:** O atendimento extraclasse deve ser solicitado e confirmado por e-mail enviado ao respectivo(s) professor(es).

## 11.2 Canal de Interação

Esta disciplina também irá prever um canal digital permanente de interação entre aluno-aluno e/ou aluno-professor para tirar dúvidas. Além disto, espera-se que este seja um fórum de interação e socialização entre os alunos o que visa permitir uma troca de experiências no processo de aprendizagem.

- **Plataforma:** Slack
- **Link:** [https://join.slack.com/t/pooqs2020ufabc/shared\\_invite/zt-h5wrahdg-TqxrF\\_oRCP0iEoMimjq8Tg](https://join.slack.com/t/pooqs2020ufabc/shared_invite/zt-h5wrahdg-TqxrF_oRCP0iEoMimjq8Tg)

O processo de ambientação e uso da ferramenta é de responsabilidade única e exclusiva do aluno, sendo o professor isento deste processo.

## 12 Informações adicionais

Os seguintes elementos relacionados com essa disciplina serão disponibilizados tanto no (Ambiente Virtual de Aprendizagem) AVA, Moodle (<https://moodle.ufabc.edu.br/>), curso “2020.SProgramação Orientada a Objetos”, quanto no link (<https://moodle.ufabc.edu.br/course/view.php?id=582>):

- Slides;
- Livros e/ou Apostilas;
- Lista de Exercícios/Atividades;
- Notas;
- Avisos.

**IMPORTANTE!** É de responsabilidade do aluno, solicitar sua inclusão no AVA, caso o mesmo deseje receber as notificações da disciplina.

## 13 Bibliografia

Em vista ao estudo e aprofundamento no conteúdo dessa disciplina, recomenda-se a seguinte bibliografia.

### 13.1 Básica

1. Booch, Grady; Rumbaugh, James; Jacobson, Ivar. ?UML: guia do usuário?. 1A edição, Editora Campus, 2006.
2. Guedes, Gilleanes T. A.. ?UML 2: uma abordagem prática?. 1a edição, Editora Novatec, 2009.
3. Deitel, Harvey M.; Deitel, Paul J. ?Java: como programar?. 6a edição, Editora Bookman, 2005
4. Caelum. Java e Orientação a Objetos. <https://www.caelum.com.br/apostila-java-orientacao-objetos/>

### 13.2 Complementar

1. Barnes, David J.; Kolling, Michael. ?Programação orientada a objetos com Java?. 4a edição, Editora Pearson Prentice Hall, 2009.
2. Flanagan, David. ?Java: o guia essencial?. 5a edição, Editora Bookman, 2006.
3. Bruegge, Bernd; Dutoit, Allen H.. ?Object-oriented software engineering: using UML, Patterns, and Java?. 2a edição, Editora Prentice Hall, 2003.
4. Larman, Craig. ?Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo?. 3a edição, Editora Bookman, 2007.
5. Fowler, Martin. ?UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos?. 3a edição, Editora Bookman, 2005.