



MC302 – Programação Orientada a Objetos

Instituto de Computação - UNICAMP

1o. Semestre de 2016

Profa. Cecília M.F.Rubira, Turmas A,B,C,D

sala 13, cmrubira@ic.unicamp.br

Versão 2.0 25/2/2016

1. Programa da Disciplina

A ementa da disciplina abrange os conceitos básicos e avançados da programação orientada a objetos, incluindo a modelagem orientada a objetos usando UML.

Os seguintes conceitos serão explorados: objetos, classes concretas e classes abstratas, mensagens, interface pública de classes, visibilidades de atributos e operações, visibilidade pública, visibilidade privada, visibilidade protegida, visibilidade de pacotes, hierarquias de generalização/especialização, hierarquias de agregação/decomposição, herança simples, herança múltipla, derivação pública, derivação privada, derivação protegida, polimorfismo de inclusão, polimorfismo paramétrico, sobrecarga (overloading), redefinição de operações (overriding), acoplamento dinâmico, interfaces versus classes, pacotes, componentes de software, tratamento de exceções, metaclasses, delegação, padrões de projeto, classes internas, enumerações, arquivos.

2. Horário das Aulas

terças e quintas das 10-12:00 (aulas teóricas)

terças das 14-16:00 turmas A e B (aulas práticas) (CC02 CC03)

terças das 16-18:00 turmas C e D (aulas práticas) (CC02 CC03)

3. Horário de Atendimento e PEDs/PAD

PED C: João Paulo Scallet joao.scalett@gmail.com

PED C: Marleny Luque Carbajal marlenyluque@gmail.com

PAD: Alexandre Medeiros alexandre.n.medeiros@gmail.com

Horário de Atendimento: combinar com a classe um horário adequado para o atendimento com PEDs e PAD. Alunos podem marcar horário com a professora para atendimento/esclarecimento de dúvidas enviando uma mensagem.

4. Critério de Avaliação

Prova1 vale 2.0.

Prova2 vale 4.0.

Labs vale 2.0.

Projeto vale 1.0.

Participação 1.0

A média M será calculada por:

Se $(Prova1 + Prova2) \geq 3.0$

$M := Prova1 + Prova2 + Labs + Projeto + Participa$

Se $(Prova1 + Prova2) < 3.0$

$M := (Prova1 + Prova2)$

Atenção: Note que as notas dos 'Labs' e do 'Projeto' só serão somadas à média final 'M' caso o aluno obtenha pelo menos 3 pontos na soma das notas da Prova1 e Prova2.

Deverão fazer exame, os alunos cuja média:

$M < 5.0$

5. Sobre o Exame de MC302

O aluno poderá fazer o exame se:

- tiver a frequência mínima exigida pela UNICAMP e
- tiver a nota mínima de 2.5 de acordo com as regras do manual do aluno.

A média do exame é:

$M_{Exame} := (M + Nota_{Exame})/2$

Se M_{Exame} for maior ou igual 5.0, o aluno está aprovado; caso contrário, reprovado.

6. Sobre os labs de MC302

- As especificações dos laboratórios estarão disponíveis no Ensino Aberto.
- Os laboratórios são individuais.
- Ao final de cada aula de laboratório, cada aluno deve submeter o laboratório, de acordo com a especificação pedida.
- A submissão do lab será feita na ferramenta portfólio do aluno, na área da disciplina no Ensino Aberto.
- A aula do lab é presencial e o aluno deve assinar a lista de presença.
- A nota do laboratório será dada: (i) se o aluno assinar a lista e (ii) se o aluno submeter o laboratório no Ensino Aberto.
- Do número total de laboratórios, serão contabilizadas as 13 maiores notas do aluno.

7. Datas Importantes

- Prova1: 26/4 (ter).
- Prova2: 21/6 (ter).
- Exame: 12/7 (ter).

8. Cronograma Detalhado da Disciplina

01/3 (ter): aula01 introdução, lab01(introdutório)
03/3 (qui): aula02 objetos e classes concretas
08/3 (ter): aula03 tipos abstratos de dados lab02(Objetos&Classes)
10/3 (qui): aula04 ciclo de vida,herança simples
15/3 (ter): aula05 agregação, padrão composite lab03(TAD)
17/3 (qui): aula06 herança múltipla
22/3 (ter): aula07 introdução ao polimorfismo lab04(Herança)
24/3 (qui): feriado -- Páscoa
29/3 (ter): aula08 polimorfismo paramétrico, acoplamento dinâmico lab03(Agregação)
31/3 (qui): aula09 polimorfismo de inclusão, classes abstratas
05/4 (ter): aula10 classes abstratas lab05(Herança Múltipla)
07/4 (qui): aula11 padrões Strategy e Observer
12/4 (ter): aula12 exercícios lab06(Pol&Acoplamento)
14/4 (qui): aula13 classes internas e projeto da disciplina
19/4 (ter): aula14 aula de exercícios lab07(Classes Abstratas)
21/4 (qui): feriado -- Tiradentes
26/4 (ter): aula15 prova1 lab08 (entrega1 projeto da disciplina)

28/4 (qui): aula17 interfaces + pacotes
03/5 (ter): aula18 pacotes + padrão Façade, lab09(Interfaces)

05/5 (qui): aula19 delegação
10/5 (ter): aula20 delegação + padrão State lab10(Pacotes)
12/5 (qui): aula21
17/5 (ter): aula 22 avaliação de curso (horário 14:00)- não tem lab
19/5 (qui): aula23
24/5 (ter): aula24, lab11
26/5 (qui): feriado -- Corpus Christi
31/5 (ter): aula25, lab12
02/6 (qui): aula26
07/6 (ter): aula27, lab13
09/6 (qui): aula28
14/6 (ter): aula29, lab14
16/6 (qui): aula30
21/6 (ter): aula31, prova2, lab15(entrega2 projeto da disciplina)
23/6 (qui):

04-08 de julho: Semana de Estudos
12 de julho (ter): Exame

9. Livros Textos

C.M.F.Rubira & P.A.C.Guerra, *Introdução à Programação Orientada a Objetos usando Java*, apostila, IC-Unicamp, 2011.

A.A.Cesta & C.M.F.Rubira, *Tutorial: A Linguagem de Programação Java*, 1996, atualizado em 2009, <http://www.ic.unicamp.br/~cmrubira/JAVATUT14PDF.pdf>

Ken Arnold, James Gosling, David Holmes, *The Java Programming Language*, Fourth Edition, Addison-Wesley Professional, 2005, ISBN 0321349806

10. Bibliografia Recomendada

Conceitos de Linguagens de Programação

Ghezzi, Carlo & Jazayeri, Mehdi, *Programming Languages Concepts*, Wiley, 3rd edition, 1997.

Sebesta, Robert W. *Concepts of Programming Languages*, Benjamin/Cummings, sixth edition, 2007.

Watt, David A. *Programming Language Concepts and Paradigms*, Prentice-Hall, 1990.

Pratt, T.W. *Programming Languages: Design and Implementation*, 2nd edition, Prentice Hall, 1983.

Orientação a Objetos

Bertrand Meyer, *Object-Oriented Software Construction*, Prentice-Hall, 1988.

Ken Arnold, & James Gosling, *Programando em Java*, Makron Books, 1997.

Ken Arnold, James Gosling, David Holmes, *A Linguagem de Programação Java*, Quarta Edição, Bookman, 2007.

Bruce Eckel, *Thinking in Java*, 4th edition, Prentice Hall, 2006.

David Flanagan, *Java in a Nutshell*, O'Reilly & Associates, 5th edition, 2005.

David Flanagan, *Java o Guia Essencial*, O'Reilly & Associates, 2006.

Rogers Cadenhead & Laura Lemay, *Sams Teach Yourself Java 2 in 21 Days*, Sams, 4th edition, 2004.

Gary Cornell & Cay S. Horstmann, *Core Java*, SunSoft Press, 1997.

Cay S. Horstmann & Gary Cornell, *Core Java 2: Volume I-Fundamentals*, Prentice Hall PTR, 2001.

Harvey M.Deitel & Paul J.Deitel, *Java How to Program*, 9th Edition, Deitel&Associates, 2012.

David J. Barnes, *Object-Oriented Programming with Java: An Introduction*, Prentice Hall, 2000.

Design Patterns

S.J. Metsker, *Padrões de Projeto em Java*, 2004, Bookman.

S.J. Metsker, *Design patterns Java workbook*, Boston : Addison-Wesley, 2002, E-book da BC-UNICAMP:

<http://proquest.safaribooksonline.com/?uiCode=unicamp&xmlId=0201743973>

M. Grand, *Patterns in Java: Volume 1*, Wiley, 1998.

E. Gamma et al., *Design Patterns: Elements of reusable OO Software*, Addison-Wesley, 1995.

Pattern Languages of Program Design, 1, 2 , 3 e 4, Software Patterns Series.

M. Fowler, *Analysis Patterns: Reusable Object Models*, Addison-Wesley, 1997.

11. Links Importantes

Thinking in Java, 3rd edition:

<http://www.mindviewinc.com/downloads/TIJ-3rd-edition4.0.zip>

Tutorial Java:

<http://www.ic.unicamp.br/~cmrubira/JAVATUT14PDF.pdf>

Ensino Aberto: www.ggte.unicamp.br/ea/

Impresso em 1 de março de 2016