



UNIVERSIDADE FEDERAL DO ABC – UFABC
CENTRO DE MATEMÁTICA, COMPUTAÇÃO E COGNIÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

PLANO DE ENSINO

ANO LETIVO	QUADRIMESTRE	TURNO	CAMPUS
2022	Q2	Diurno	Santo André

CÓDIGO	NOME
MCTA018-13SA	Programação Orientada a Objetos
Turma	RECOMENDAÇÃO
DA2MCTA018-13SA	Processamento da Informação Programação Estruturada

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.

SOBRE A DISCIPLINA

O material da disciplina será disponibilizado no ambiente virtual de aprendizagem (AVA) Moodle UFABC: <https://moodle.ufabc.edu.br/>

Curso: DA2MCTA018-13SA - Programação Orientada a Objetos - Paulo Henrique Pisani - 2022.2

Fique atento aos comunicados realizados no Moodle UFABC.

Mapa de Atividades

Disciplina: Programação Orientada a Objetos – T-P-I 2-2-4

Docente: Paulo Henrique Pisani

Quadrimestre: 2022.Q2 (terça-feira das 8h às 10h semanal; quinta-feira das 10h às 12h semanal)

Descrição das atividades no mapa de atividades:

- **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.
- **Exercícios para submissão pelo Moodle:** serão disponibilizados exercícios ao longo do quadrimestre. Todos os exercícios são individuais (devem ser resolvidos pelo próprio aluno). Os exercícios deverão ser submetidos pelo Moodle. Cada exercício pode ter pontuação máxima e prazo de entrega diferentes (essas informações estarão em cada exercício no Moodle). Os exercícios de programação (EPs) para nota possuem a indicação **[EP]** no Moodle. Também haverá um exercício para nota bônus, que possuirá a indicação **[Bonus]** no Moodle.
- **Assistir aulas:** aulas presenciais, realizadas na sala de aula e no laboratório.
- **Avaliações P1, P2, Substitutiva e Recuperação:** avaliações presenciais síncronas realizadas durante o horário da aula.

Avaliação dos exercícios de programação (EPs) e Bônus:

Poderá ser utilizado um sistema de correção automática. A princípio, a correção automática (caso esteja disponível) aplicará o seguinte critério:

- *Exercícios com pontuação máxima até 1 ponto:* é necessário passar em todos os casos de teste para obter 1 ponto. Caso contrário, ficará com zero.
- *Exercícios com pontuação máxima superior a 1 ponto:* se não passar nos casos de teste abertos (que tem entrada e saída disponíveis/abertas), ficará com zero. Se passar nos casos de teste abertos, a nota poderá ser maior que zero. Dessa forma, estes exercícios podem receber pontuação parcial.
- **Importante:** o professor poderá atribuir nota manualmente nos exercícios e alterar a nota atribuída pelo sistema de correção automática também (por exemplo, devido a pontos que o corretor automático não pode identificar ou problemas/erros na avaliação).

Semana	Horas (T+P)	Tema / Subtema	Objetivos específicos	Atividades
01	4h	<p>Introdução ao Java</p> <ul style="list-style-type: none"> • Tipos de dados • Entrada/Saída • Operadores aritméticos, relacionais, lógicos • Estrutura condicional • Estruturas de repetição • Vetores e matrizes • Métodos • Estilo de codificação 	<p>Aplicar a linguagem Java para implementação de programas, incluindo os programas propostos na disciplina.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
02	2h	<p>Classes, objetos, encapsulamento, UML</p> <ul style="list-style-type: none"> • Classes, atributos e métodos • Classe vs Instância • Modificadores de acesso • Pacotes • UML (representação de uma classe) • UML (relacionamentos: associação, agregação e composição) 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
03	4h	<p>Construtores, sobrecargas</p> <ul style="list-style-type: none"> • Construtores • Sobrecargas 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>

04	4h	<p>Herança, polimorfismo</p> <ul style="list-style-type: none"> • Herança • Polimorfismo • UML (herança no diagrama de classes) 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
05	4h	<p>Interfaces, polimorfismo</p> <ul style="list-style-type: none"> • Classe abstrata • Interfaces • UML (interface no diagrama de classes) 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
06	4h	Avaliação P1	Avaliar as competências adquiridas.	<p>- Assistir aula (revisão)</p> <p>- Avaliação P1</p>
07	4h	<p>Construtores, cópia de objetos, final</p> <ul style="list-style-type: none"> • Chamada a outros construtores • Acesso a construtores • Cópia de objetos • Modificador <i>final</i> 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>

08	4h	<p>Padrões de projeto de software</p> <p>Estruturas lineares no contexto de orientação a objetos</p> <ul style="list-style-type: none"> • Modificadores de acesso em classes • Aplicação dos conceitos vistos em estruturas lineares 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
09	4h	<p>Tratamento de exceções</p> <ul style="list-style-type: none"> • Hierarquia de exceções • Estruturas para lançar e capturar exceções 	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
10	4h	<p>Generics</p>	<p>Entender os conceitos de programação orientada a objetos apresentados até esta semana.</p> <p>Aplicar a linguagem Java nos problemas propostos na disciplina, implementando com uso dos conceitos de programação orientada a objetos apresentados até esta semana.</p>	<p>- Assistir aulas</p> <p>- Exercícios para submissão pelo Moodle.</p>
11	4h	<p>Avaliação P2</p>	<p>Avaliar as competências adquiridas.</p>	<p>- Assistir aula (revisão)</p> <p>- Avaliação P2</p>
12	4h	<p>Avaliação Substitutiva</p>	<p>Avaliar as competências adquiridas.</p>	<p>- Avaliação Substitutiva</p>
Reposição	2h	<p>Recuperação</p>	<p>Avaliar as competências adquiridas.</p>	<p>- Avaliação de Recuperação</p>

PLANEJAMENTO PRELIMINAR DE AULAS

As aulas teóricas ocorrerão às 8h e as aulas práticas às 10h. Os dias das aulas estão indicados no calendário a seguir.

1	07/06/2022	terça-feira	Teoria	Apresentação inicial, Introdução ao Java
2	09/06/2022	quinta-feira	Prática	Introdução ao Java
3	14/06/2022	terça-feira	Teoria	Classes, objetos, encapsulamento, UML
4	21/06/2022	terça-feira	Teoria	Construtores, sobrecargas
5	23/06/2022	quinta-feira	Prática	Classes, objetos, encapsulamento, construtores, sobrecargas
6	28/06/2022	terça-feira	Teoria	Herança, polimorfismo
7	30/06/2022	quinta-feira	Prática	Herança, polimorfismo
8	05/07/2022	terça-feira	Teoria	Interfaces, polimorfismo
9	07/07/2022	quinta-feira	Prática	Interfaces, polimorfismo
10	12/07/2022	terça-feira	Teoria	Revisão
11	14/07/2022	quinta-feira	Prática	Avaliação P1
12	19/07/2022	terça-feira	Teoria	Construtores, cópia de objetos, final
13	21/07/2022	quinta-feira	Prática	Construtores, cópia de objetos, final
14	26/07/2022	terça-feira	Teoria	Padrões de projeto de software, Estruturas lineares no contexto de orientação a objetos
15	28/07/2022	quinta-feira	Prática	Padrões de projeto de software, Estruturas lineares no contexto de orientação a objetos
16	02/08/2022	terça-feira	Teoria	Tratamento de exceções
17	04/08/2022	quinta-feira	Prática	Tratamento de exceções
18	09/08/2022	terça-feira	Teoria	Generics
19	11/08/2022	quinta-feira	Prática	Generics
20	16/08/2022	terça-feira	Teoria	Revisão
21	18/08/2022	quinta-feira	Prática	Avaliação P2
22	23/08/2022	terça-feira	Teoria	Vista de avaliação
23	25/08/2022	quinta-feira	Prática	Avaliação Substitutiva
24	29/08/2022	segunda-feira	Prática	Avaliação de Recuperação

AVALIAÇÃO

Avaliação do Período Letivo Regular:

A média final será composta por:

- **Avaliação P1 (peso 35%):** 14/07/2022 às 10h (avaliação presencial, realizada durante o horário da aula). Será atribuída uma nota de 0 a 10 na Avaliação P1.
- **Avaliação P2 (peso 35%):** 18/08/2022 às 10h (avaliação presencial, realizada durante o horário da aula). Será atribuída uma nota de 0 a 10 na Avaliação P2.
- **Média de Exercícios de programação (peso 30%):** média das notas obtidas em todos os exercícios de programação (EPs) disponibilizados no Moodle (a média será uma nota de 0 a 10). Ao longo do quadrimestre, serão disponibilizados diversos EPs: cada exercício pode ter pontuação máxima diferente assim como prazos de entrega diferentes. Os exercícios de programação para nota possuem a indicação **[EP]** no Moodle.
- **Bônus:** um exercício bônus que vale até 0,5 ponto na média final. Esse exercício possui a indicação **[Bonus]** no Moodle.

Exercícios não entregues no prazo receberão nota zero.

Importante: Todos os exercícios são individuais (devem ser resolvidos pelo próprio aluno). Será utilizada ferramenta para identificação de submissões com alta similaridade no Moodle. Caso seja identificado plágio em qualquer exercício entregue, será atribuída nota zero a todos os envolvidos.

A Média de Exercícios de Programação (M_{EPs}) será obtida da seguinte forma:

$$M_{EPs} = \left(10 \times \frac{\text{soma da pontuação obtida nos EPs}}{\text{soma da pontuação máxima dos EPs}} \right)$$

A Média Final (MF) será obtida da seguinte forma:

$$MF = (0,35 \times P1) + (0,35 \times P2) + (0,3 \times M_{EPs}) + Bônus$$

Observação: o item Bônus valerá no máximo 0,5 ponto na média final (MF).

A partir da média final (MF), o conceito final (CF) será obtido por meio da seguinte tabela:

Conceito Final (CF)	Média Final (MF)
A	$MF \geq 9$
B	$9 > MF \geq 7,5$
C	$7,5 > MF \geq 6$
D	$6 > MF \geq 5$
F	$MF < 5$
O	(reprovação por ausência de avaliação)

Avaliação Substitutiva (SUB):

Estarão habilitados para a avaliação substitutiva os alunos que se ausentarem a uma das avaliações do período regular (P1 ou P2) e contemplados pelo benefício de acordo com a Resolução CONSEPE no. 227, de 23 de abril de 2018. Nesta hipótese, o aluno deverá apresentar uma justificativa válida para o docente pelo menos 24h antes do dia da avaliação substitutiva. A nota da avaliação SUB substitui a nota de P1 ou de P2.

Data da Avaliação Substitutiva (SUB): 25/08/2022 às 10h (avaliação presencial, realizada durante o horário da aula)

Mecanismo de Recuperação (REC):

Estarão habilitados para o mecanismo de recuperação os alunos que obtiverem conceito final **D** ou **F**, obedecendo as regras indicadas na Resolução CONSEPE no. 182 de 23 de outubro de 2014.

Data da Avaliação de Recuperação (REC): 29/08/2022 às 10h (avaliação presencial, realizada durante o horário da aula)

A média final com recuperação é obtida da seguinte forma, em que REC é a nota (0 a 10) obtida na Avaliação de Recuperação:

$$MR = (0,5 \times MF) + (0,5 \times REC)$$

O Conceito Final com Recuperação é obtido conforme tabela a seguir (o conceito final com recuperação será no máximo C):

Conceito Final com Recuperação (CFR)	Média Final com Recuperação (MR)
C	MR ≥ 6
D	6 > MR ≥ 5
F	MR < 5

FERRAMENTAS

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

- Navegador web compatível com o Moodle UFABC. O navegador web deve permitir a execução de vídeos no Eduplay (<https://eduplay.rnp.br/>).
- Editor de texto (por exemplo, gedit, Notepad++, etc).
- Java Development Kit: necessário para desenvolvimento de programas em Java - é recomendado o Open Java Development Kit (OpenJDK) 11.

ATIVIDADES DE APOIO

Esta disciplina prevê um horário de atendimento extraclasse para atividades de apoio aos estudantes regulares desta turma, conforme disposto na Resolução CONSUNI nº 183, de 31 de outubro de 2017.

O horário de atendimento semanal será realizado no seguinte dia, local e horário:

- **terça-feira, das 11h às 13h**, sala 507-2, bloco A, Santo André (Prof. Paulo)

BIBLIOGRAFIA

Bibliografia básica:

- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. Rio de Janeiro, RJ: Campus, 2005.
- DEITEL, H. M.; DEITEL, P. J. Java: como programar. 6. ed. Porto Alegre, RS: Bookman, 2005.
- GUEDES, G. T. A. UML 2: uma abordagem prática. São Paulo, SP: Novatec, 2009.

Bibliografia complementar:

- BARNES, D. J.; KOLLING, M. Programação orientada a objetos com Java. 4. ed. São Paulo, SP: Pearson Prentice Hall, 2009.
- BRUEGGE, B.; DUTOIT, A. H. Object-oriented software engineering: using UML, patterns, and Java. 2. ed. Upper Saddle River, NJ: Prentice Hall, 2003.
- FLANAGAN, D. Java: o guia essencial. 5. ed. Porto Alegre, RS: Bookman, 2006.
- FOWLER, M. UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos. 3. ed. Porto Alegre, RS: Bookman, 2005.
- LARMAN, C. Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo. 3. ed. Porto Alegre, RS: Bookman, 2007.

PROFESSOR RESPONSÁVEL

Prof. Dr. Paulo Henrique Pisani