

<b>Campus:</b> São José dos Campos/ICT		
<b>Curso(s):</b> BCC/EC/BCT		
<b>Unidade Curricular (UC):</b> Programação Orientada a Objetos		
<b>Unidade Curricular (UC):</b> <i>Object-Oriented Programming</i>		
<b>Unidade Curricular (UC):</b> <i>[nome da UC em espanhol - opcional]</i>		
<b>Código da UC:</b> 2471		
<b>Docente Responsável/Departamento:</b> Rodrigo Colnago Contreras / DCT		<b>Contato (e-mail):</b> contreras@unifesp.br
<b>Docente (s) Colaborador/a (es/as)/ Departamento(s):</b>		<b>Contato (e-mail):</b> [opcional]
<b>Ano letivo:</b> 2025	<b>Termo:</b> 2	<b>Turno:</b> I
<b>Nome do Grupo/Módulo/Eixo da UC:</b> (se houver)		<b>Idioma predominante em que a UC será oferecida:</b> <input checked="" type="checkbox"/> Português <input type="checkbox"/> English <input type="checkbox"/> Español <input type="checkbox"/> Français <input type="checkbox"/> Libras <input type="checkbox"/> Outro:
<b>UC:</b> <input checked="" type="checkbox"/> Fixa <input type="checkbox"/> Eletiva <input type="checkbox"/> Optativa	<b>Oferecida como:</b> <input checked="" type="checkbox"/> Disciplina <input type="checkbox"/> Módulo <input type="checkbox"/> Estágio <input type="checkbox"/> Outro:	<b>Oferta da UC:</b> <input checked="" type="checkbox"/> Semestral <input type="checkbox"/> Anual
<b>Ambiente Virtual de Aprendizagem:</b> <input type="checkbox"/> Moodle <input type="checkbox"/> Classroom <input type="checkbox"/> Não se aplica <input checked="" type="checkbox"/> Outro: Discord		
<b>Pré-Requisito(s) – Indicar código e nome(s) da(s) UC(s):</b> Algoritmos e Estruturas de Dados 1		
<b>Carga horária total (em horas):</b> 72h		
<b>Carga horária teórica:</b> 36h	<b>Carga horária prática:</b> 36h	<b>Carga horária de extensão</b> (se houver):
<b>Se houver atividades de extensão, indicar código e nome do projeto ou programa vinculado na Pró-Reitoria de Extensão e Cultura (ProEC):</b>		
<b>Ementa:</b> Introdução à Programação Orientada a Objetos; Classes e Métodos; Encapsulamento e Sobrecarga; Sobreposição de Métodos; Construtores e Destrutores; Herança; Polimorfismo e Ligação Dinâmica; Introdução a uma linguagem OO (Tipos de Dados, Operadores, Variáveis, Arrays, Controle de Fluxo); Tratamento de Exceção; Manipulação de Strings e Arquivos; Aplicações; Estudos de Caso.		
<b>Conteúdo programático:</b>  1. Introdução à programação orientada a objetos 2. Classes, Métodos e Atributos 3. Construtores e sobrecarga 4. Atributos e métodos estáticos 5. Estruturas de controle e decisão 6. Reutilização de classes (Herança) 7. Classes abstratas e interfaces 8. Pacotes de classes 9. Arrays e Matrizes 10. Classes de manipulação de strings e arquivos 11. Coleções de objetos 12. Tratamento de Exceção		
<b>Objetivos:</b> <u>Gerais:</u> Capacitar o aluno para o desenvolvimento de software orientado a objetos, utilizando uma linguagem de programação com grande aceitação no meio comercial e acadêmico.  <u>Específicos:</u> Propiciar ao aluno uma adaptação (transição) entre a programação estruturada/procedimental e a programação orientada a objetos; Projetar, implementar, testar e depurar programas orientados a objetos; Introduzir os conceitos de		

classes e objetos, herança e polimorfismo; e Apresentar uma visão geral dos recursos avançados da linguagem.

**Metodologia de ensino:** A disciplina será intercalada por aulas teóricas e aulas práticas em laboratório. Nas aulas teóricas serão apresentados os principais conceitos e seus relacionamentos. Já nas aulas de laboratório, os conceitos serão implementados principalmente em linguagens Java, Python e/ou C#, utilizando-se ferramentas de codificação e testes baseados em software livre. Ademais, desenvolver-se-á atividades à distância, com o apoio da ferramenta Classroom.

A metodologia de ensino baseada na resolução de problemas (*Problem Based Learning*) será amplamente utilizada. O professor, após apresentar a teoria necessária, irá propor problemas e atuará apenas como facilitador junto aos alunos na resolução do problema.

**Avaliação:**

Serão adotados os seguintes instrumentos de avaliação:

- 2 provas escritas (P1 e P2).

A nota final será calculada da seguinte forma:  $NF = 0.5 * (P1 + P2)$

A promoção do aluno na UC obedecerá aos critérios estabelecidos pela Pró-Reitoria de Graduação, tal como discutido no projeto pedagógico do curso.

**Bibliografia:**

**Básica:** Deitel, Harvey M., and Paul J. Deitel. *Cómo programar en Java*. Pearson educación, 2003.

Budd, Timothy. *Understanding object-oriented programming with Java*. Addison-Wesley Longman Publishing Co., Inc., 1998.

Deitel, Harvey M., and Paul J. Deitel. *C++ how to program*. Pearson Educación, 2003.

Sempf, Bill, Charles Sphar, and Stephen R. Davis. *C# 2010 All-in-one for Dummies*. John Wiley & Sons, 2010.

**Complementar:** Videoaulas, artigos científicos e demais materiais complementares serão disponibilizados na página do curso.

**Cronograma:** [opcional]