

Campus: São José dos Campos		
Curso (s): Engenharia de Computação e Ciência da Computação		
Unidade Curricular (UC): Tópicos – Programação Funcional		
Unidade Curricular (UC): [nome da UC em inglês]		
Código da UC:		
Docente Responsável: Otávio Augusto Lazzarini Lemos		Contato (e-mail): otavio.lemos@unifesp.br
Docente (s) Colaborador/a (es/as):		Contato (e-mail): [opcional]
Ano letivo:	Termo:	Turma (s):
Nome do Grupo/Módulo/Eixo da UC (se houver):		Idioma predominante em que a UC será oferecida: <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:
UC: <input type="checkbox"/> Fixa <input checked="" type="checkbox"/> Eletiva <input type="checkbox"/> Optativa	Oferecida como: <input type="checkbox"/> Disciplina <input type="checkbox"/> Módulo <input type="checkbox"/> Estágio <input type="checkbox"/> Outro:	Oferta da UC: <input type="checkbox"/> Semestral <input type="checkbox"/> Anual
Ambiente Virtual de Aprendizagem: <input type="checkbox"/> Moodle <input checked="" type="checkbox"/> Classroom <input type="checkbox"/> Outro: <input type="checkbox"/> Não se aplica		
Pré-Requisito (s) - Indicar Código e Nome (s) da (s) UC: AED I (FORTEMENTE RECOMENDADO)		
Carga horária total (em horas): 72		
Carga horária teórica (em horas): 36	Carga horária prática (em horas): 36	Carga horária de extensão (em horas, se houver):
Se houver atividades de extensão, indicar código e nome do projeto ou programa vinculado na Pró-Reitoria de Extensão e Cultura (ProEC):		
<p>Ementa: Características da Programação Funcional; Programação Funcional na Prática com Clojure:</p> <p>A Floresta das Ferramentas Um ambiente de programação amigável e eficiente facilita a experimentação de suas ideias. Você aprenderá como configurar seu ambiente.</p> <p>A Montanha da Linguagem Ao subir, você obterá conhecimento da sintaxe, semântica e estruturas de dados de Clojure. Você aprenderá como usar uma das ferramentas de programação mais poderosas, a macro, e como simplificar sua vida com as construções de simultaneidade do Clojure.</p> <p>A Caverna dos Artefatos Em suas profundezas, você aprenderá a construir, executar e distribuir seus próprios programas e como usar bibliotecas de código. Você também aprenderá a relação de Clojure com a Java Virtual Machine (JVM).</p> <p>O Castelo nas Nuvens da Mentalidade Em seu ar rarefeito, você conhecerá o porquê e o como do Lisp e da programação funcional. Você aprenderá sobre a filosofia de simplicidade que permeia Clojure e como resolver problemas como um clojurista.</p>		
<p>Conteúdo programático:</p> <p><i>Parte I: Configuração do Ambiente</i></p> <p>1: Construindo, Executando e o REPL</p> <p>2: Como usar o VS Code para programar em Clojure</p> <p><i>Parte II: Fundamentos da Linguagem</i></p> <p>3: Faça coisas: um curso intensivo de Clojure</p> <p>4: Funções essenciais em profundidade</p> <p>5: Programação Funcional</p> <p>6: Organizando seu projeto: o conto de um bibliotecário</p> <p>7: Clojure Alchemy: Leitura, Avaliação e Macros</p> <p>8: Escrevendo Macros</p> <p><i>Parte III: Tópicos Avançados</i></p>		

9: Programação Concorrente e Paralela

10: Clojure Metaphysics: Atoms, Refs, Vars e Cuddle Zombies

11: Processos simultâneos principais com `core.async`

12: Interagindo com Java

13: Crie e estenda abstrações com multimétodos, protocolos e registros

Objetivos:

[descrição da contribuição da UC para a formação do/a discente]

Gerais: Aprender a essência da programação funcional de uma maneira prática.

Específicos: Aprender os rudimentos de uma linguagem funcional para poder construir aplicações reais usando PF.

Metodologia de ensino: Aulas expositivas e práticas, em particular programando juntos, alunos e professor, em uma espécie de Mob Programming.

Avaliação:

- Propostas de exercícios semanais formando a Média dos Exercícios (ME)

- Duas provas formando a Média das Provas (MP)

- Nota Final: $0.8*MP + 0.2*ME$

(Pode ser que se opte por utilizar apenas os exercícios, sem aplicação de provas, de acordo com o nível de aprendizado dos alunos)

Bibliografia:

Básica:

1. Daniel Higginbotham. Clojure for the brave and true (2016).
<https://www.braveclojure.com/clojure-for-the-brave-and-true/>

Complementar:

1. Clojure Applied
2. Getting Clojure
3. Programming Clojure
4. Domain Modeling Made Functional
5. Functional Programming: A PragPub Anthology Exploring Clojure, Elixir, Haskell, Scala, and Swift, by Michael Swaine and the PragPub writers