



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS FÍSICAS E MATEMÁTICAS
Departamento de Física
Campus Trindade - CEP 88040-900 - Florianópolis SC
Tel: 48 3721-2876

PLANO DE ENSINO 2025.2

Em acordo com a Resolução nº 003/CEPE/8405 de Abril de 1984

I. IDENTIFICAÇÃO DA DISCIPLINA:

CÓDIGO	NOME DA DISCIPLINA	HORAS-AULA SEMANAIS		HORAS-AULA SEMESTRAIS
		TEÓRICAS	PRÁTICAS	
FSC 7114	INTRODUÇÃO À FÍSICA COMPUTACIONAL	-	4 HA	72 HA

II. PRÉ-REQUISITO(S)(Código(s) e nome da(s) disciplina(s))

FSC 5165 E FÍSICA GERAL II-A E
(MTM 5512 OU (GEOMETRIA ANALÍTICA OU
MTM 3121) ÁLGEBRA LINEAR)

III. CURSO(S) PARA O(S) QUAL(IS) A DISCIPLINA É OFERECIDA

NOME DO CURSO	TURMA	HORÁRIO
Física - Licenciatura	6225	220202/420202

IV. PROFESSOR(ES) MINISTRANTE(S)

André Luiz de Amorim

V. EMENTA

Explicitação de conceitos físicos e matemáticos em forma de algoritmos computacionais e sua implementação em alguma linguagem de alto nível compilável (C, Fortran, etc.) ou de script (JavaScript, Python, Perl, Matlab, Mathematica, Maple, etc.) com ênfase no paradigma estruturado (não orientado a objeto) mediante a utilização de variáveis numéricas e string, comandos de entrada e saída, estrutura de decisão, estruturas de repetição, matrizes e subprogramas.

VI. OBJETIVOS

Introduzir conceitos de programação estruturada e aplicar estes na resolução de problemas matemáticos simples.

VII. CONTEÚDO PROGRAMÁTICO

- O computador
 - Breve história do computador
 - Arquitetura do computador
 - Sistemas operacionais
 - Interação com o sistema operacional
 - Linguagens de programação
- Algoritmos
 - Conceitos de algoritmo
 - Pseudo-código
- Variáveis
 - Conceitos de tipagem: forte e dinâmica
 - Definição de variável: conceito de atribuição
 - Variáveis numéricas
 - Simples
 - Dupla
 - Representação científica
 - Variáveis lógicas
 - Operadores relacionais
 - Operadores lógicos
 - Expressões lógicas
 - Precedência

- Variáveis complexas (se suportado pela linguagem escolhida)
- Variáveis string
- Variáveis definidas pelo usuário (Type em Fortran ou struct em C)
- Números binários
- Operações matemáticas
 - Operações simples (adição, subtração, produto, divisão, módulo, etc.)
 - Operações complexas (funções matemáticas como as trigonométrica, logarítmicas, etc.)
 - Operações em nível de bits
 - Precedência de operadores
- Estruturas de decisão
 - Simples (if)
 - Complexa (else)
 - Específica (case)
 - Estruturas aninhadas
- Estruturas de repetição
 - Com contador (laço do em Fortran ou for em C)
 - Com expressão lógica (while)
 - Estruturas aninhadas
- Matrizes
 - Estática
 - Dinâmica (allocate em Fortran ou malloc em C)
- Entrada e saída para arquivos
 - Abertura
 - Leitura
 - Escrita
 - Uso de estruturas de repetição na leitura e/ou escrita
- Subprogramas
 - Vantagens da modularização
 - Escopo de variáveis e variáveis globais
 - Funções
 - Passagem de argumentos por valor
 - Passagem de argumentos por referência
 - Subrotinas
 - Recursividade

VIII. PROCEDIMENTOS METODOLÓGICOS / DESENVOLVIMENTO DO PROGRAMA

As atividades desta disciplina são práticas, a serem realizadas no laboratório de informática do departamento de Física durante o horário de aula.

A linguagem de programação utilizada será Python, utilizando o paradigma de *Literate Programming*, através de notebooks Jupyter. Os estudantes que desejarem utilizar outra linguagem deverão propor ao professor uma alternativa viável de entrega das avaliações.

O conteúdo do semestre será dividido em semanas, conforme o cronograma informado pelo professor. No início das aulas, o conteúdo será apresentado de forma expositiva, usando projeção da tela de computador do professor e quadro branco. No restante da aula, os estudantes irão resolver problemas propostos. As notas de aula serão disponibilizadas no Moodle em formato de jupyter notebooks.

Aproximadamente a cada duas semanas haverá avaliações com problemas a serem programados em sala de aula.

O estudante deverá idealizar, desenhar e resolver um problema de física computacional paralelamente ao curso. As últimas semanas do semestre serão destinadas ao desenvolvimento do projeto.

IX. ATIVIDADES PRÁTICAS (se houver)

Todas as atividades deste curso serão desenvolvidas em computador, onde o aluno irá escrever programas para resolver problemas propostos pelo professor.

X. FORMAS DE AVALIAÇÃO E REGISTRO DE FREQUÊNCIA

Avaliações: O professor irá apresentar notebooks contendo problemas referentes ao conteúdo visto nas semanas anteriores, a serem programados em sala de aula, com nota entre 0 e 10. No final do semestre haverá uma nota referente às avaliações (NA), que é a média aritmética arredondada conforme os critérios da UFSC.

Projeto: O estudante deverá idealizar, desenhar e resolver um problema de física computacional, a ser entregue no final do semestre. O projeto deverá ser entregue em formato de relatório ou artigo científico. Recomenda-se entregar um arquivo Jupyter Notebook ou PDF, acompanhados de código fonte e dados auxiliares conforme o caso. O projeto receberá uma nota de 0 a 10 (NP).

Frequência: O aluno obterá frequência suficiente (FS) caso frequente 75% das aulas ou mais. Caso a frequência seja menor que 75%, o aluno terá frequência insuficiente (FI).

Média final: A média do semestre (MS) será a média aritmética da nota das avaliações (NA) e da nota do projeto (NP), arredondada conforme os critérios da UFSC.

Aprovação: Os alunos com FS, e $MS \geq 6,0$ serão considerados aprovados.

Reposição justificada: Não haverá prova de recuperação, por se tratar de uma disciplina prática. Avaliações perdidas devido a faltas justificadas poderão ser repostas na semana de reposição. Para entregar uma justificativa, o estudante deverá utilizar o formulário disponível no site do Departamento de Física.

XI. LIMITES LEGAIS DO DIREITO DE AUTOR E IMAGEM (em acordo com a Lei nº 9.610/98–Lei de Direitos Autorais)

A legislação pertinente será observada (lei 9.610/98).

• XII. ATENDIMENTO AO ESTUDANTE (horário/monitoria – se houver)

Atendimento: Segunda-feira 17h–18h; quarta-feira 17h–18h.

Monitoria: Ver horários afixados na porta do laboratório de Informática.

XIII. REFERÊNCIAS BIBLIOGRÁFICAS (Básica e Complementar)

- Silva, N. C. da, Física com Javascript, Universidade Federal de Santa Catarina, 2ª edição (2016).
 - Disponível em <http://canzian.fsc.ufsc.br/fisicacomjavascript/>
- Refsnes Data (w3Schools), JavaScript Tutorial (2020).
 - Disponível em <https://www.w3schools.com/js/default.asp>
- Mozilla Foundation, MDN Web Docs - JavaScript (2020).
 - Disponível em <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- Khan Academy, Intro to JS: drawing and animation (2020).
 - Disponível em <https://www.khanacademy.org/computing/computer-programming/programming>

Bibliografia complementar

- Notas de aula do professor, disponíveis no ambiente Moodle.
- Chabay, Ruth W. e Sherwood, Bruce A., Matter and interactions, Wiley, 2010.
- Ellis, T. M. R., Philips, I. R., Lahey, Thomas M., Fortran 90 Programming, Addison-Wesley, 1994.
- Shildt, Herbert. C, completo e total (3ª Ed.), Makron, 1997.
- Menezes, Nilo N, C., Introdução à programação com Python: algoritmos e lógica de programação para iniciantes, Novatec, 2010.
- Morrisson, Michael, Use a cabeça: JavaScript, Alta Books, 2008.
- Stallings, William, Arquitetura e organização de computadores, Pearson, 2010.

XIV. CRONOGRAMA

Semana – Conteúdo

- 1 – História do computador / familiarização com o laboratório
- 2 – Jupyter notebooks
- 3 – Scripts Python e terminal interativo
- 4 – Variáveis, operadores e funções matemáticas
- 5 – Representação numérica em computadores: lógica e aritmética binária

- 6 – Strings
- 7 – Listas, dicionários e outros contêineres
- 8 – Entrada e saída de dados
- 9 – Controle de fluxo: if-else-if-else
- 10 – Controle de fluxo: laços de repetição e iteradores
- 11 – Vetores e matrizes
- 12 – Gráficos usando matplotlib
- 13 – Funções
- 14 – Projeto
- 15 – Projeto
- 16 – Projeto
- 17 – Projeto
- 18 – Entrega do projeto e reposição justificada

O cronograma é indicativo, e poderá sofrer alterações. O conteúdo programático poderá ser trabalhado em ordem diferente da proposta.