



**MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS FÍSICAS E MATEMÁTICAS
DEPARTAMENTO DE FÍSICA**

**Plano de ensino
para ensino remoto emergencial em 2020-1**

FSC 7114 - Introdução à Física Computacional

1. Identificação da disciplina

Nome: Introdução à Física Computacional

Código: FSC 7114

Pré-requisito: FSC5165 e MTM5512

Carga horária: 72 ha (4 ha/semana)

Semestre letivo: 2020-1

Professores:

André Luiz de Amorim

Nelson Canzian da Silva

Cursos:

Bacharelado em Física

Licenciatura em Física

Bacharelado em Meteorologia

Turmas e horários:

03002 e 03230: 3a. 09:20 (2 aulas) e 5a. 09:20 (2 aulas)

04225: 4a. 20:20 (2 aulas) e 6a. 20:20 (2 aulas)

2. Objetivo Geral

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

3. 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.

4. Programa

A. O computador

1. Breve história do computador

2. Arquitetura do computador

3. Sistemas operacionais

4. Interação com o sistema operacional

5. Linguagens de programação

B. Algoritmos

1. Conceito de algoritmo
2. Pseudo-código

C. Variáveis

1. Conceito de tipagem: forte e dinâmica
2. Definição de variável: conceito de atribuição
3. Variáveis numéricas
 1. Simples
 2. Dupla
 3. Representação científica
4. Variáveis lógicas (se suportado pela linguagem escolhida)
 1. Operadores relacionais
 2. Operadores lógicos
 3. Expressões lógicas
 4. Precedência
5. Variáveis complexas (se suportado pela linguagem escolhida)
6. Variáveis string
7. Variáveis definidas pelo usuário (Type em Fortran 90-2008 ou struct em C)
8. Números binários

D. Operações matemáticas

1. Operações simples (adição, subtração, produto, divisão, módulo, etc.)
2. Operações complexas (funções matemática como as trigonométricas, logarítmicas, etc.)
3. Operações em nível de bits
4. Precedência de operadores

E. Estruturas de decisão

1. Simples (if)
2. Complexa (else)
3. Específica (case)
4. Estruturas aninhadas

F. Estruturas de repetição

1. Com contador (laço do em Fortran ou for em C)
2. Com expressão lógica (while)
3. Estruturas aninhadas

G. Matrizes

1. Estática
2. Dinâmica (allocate em Fortran 90-2008 ou malloc em C)

H. Entrada e saída para arquivos

1. Abertura
2. Leitura
3. Escrita
4. Uso de estruturas de repetição na leitura e/ou escrita

I. Subprogramas

1. Vantagens da modularização
2. Escopo de variáveis e variáveis globais
3. Funções

1. Passagem por valor
2. Passagem por referência
4. Subrotinas
5. Recursividade

5. Cronograma

As aulas remotas ocorrerão a partir de 31/08/2020, ao longo de 16 semanas, observando o Anexo da resolução normativa No 140/2020/CUn, de 21/07/2020.

- Semana 1: Operadores e funções matemáticas
- Semana 2: Caixas de diálogo
- Semana 3: Controle de fluxo
- Semana 4: Matrizes unidimensionais
- Semana 5: Matrizes multidimensionais
- Semana 6: Funções
- Semana 7: Strings
- Semana 8: HTML/CSS/JavaScript
- Semana 9: Interação
- Semana 10: Tempo
- Semana 11: Canvas
- Semana 12 a 16: Projeto

6. Metodologia

Na sua realização presencial, a disciplina é essencialmente "laboratorial". Cada encontro (1 h 40 min) inicia com uma breve (20-30 minutos) explicação sobre o conteúdo a ser explorado e a partir disso os alunos desenvolvem programas de computador que resolvem os exercícios propostos no texto de base. Nesta fase, o papel do professor é o de circular entre as estações de trabalho para motivar os alunos e resolver dúvidas pontuais. Ao final de cada bloco de exercícios, o professor fecha o tema com alguns comentários sobre os principais problemas enfrentados pelos alunos. Usualmente, é abordado um capítulo do texto de base a cada semana.

Na sua realização remota emergencial os alunos serão instados a ler um capítulo do texto de base a cada semana e resolver os exercícios (que pedem pequenas modificações em programas dados como exemplo). Uma série de vídeos instrucionais será disponibilizada para auxiliar nessa tarefa. O professor estará disponível para atendê-los nos dias e horários regulares das aulas através da ferramenta que acharem mais conveniente (Moodle, e-mail, WhatsApp etc.), ou em atendimentos exclusivos em horários acordados conjuntamente.

7. Avaliação e frequência

A média final será computada a partir dos exercícios encaminhados semanalmente (50%) e do projeto final (50%), que consistem em um pequeno problema e solução computacional proposto pelo próprio aluno. Quanto ao projeto, serão avaliados (a) dificuldade da proposta; (b) qualidade da modelagem; (c) qualidade da interface com o usuário. A frequência será proporcional ao número de exercícios entregues no prazo estipulado, inclusive o projeto final.

8. Bibliografia

Para o período emergencial, disponível on-line gratuitamente:

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

Bibliografia regular:

1. Chabay, Ruth W. e Sherwood, Bruce A. *Matter and interactions*, Wiley, 2010.
2. Ellis, T. M. R., Philips, I. R., Lahey, Thomas M. *Fortran 90 Programming*, Addison-Wesley, 1994.
3. Schildt, Herbert. *C completo e total (3a. Ed.)*, Makron, 1997.
4. Menezes, Nilo. N. C. *Introdução à programação com Python: algoritmos e lógica de programação para iniciantes*, Novatec, 2010.
5. Morrisson, Michael. *Use a cabeça: JavaScript*, Alta Books, 2008.
6. Stallings, William. *Arquitetura e organização de computadores*, Pearson, 2010.