**UNIVERSIDADE FEDERAL DE SANTA CATARINA – UFSC**
**DEPARTAMENTO DE CIÊNCIA DA INFORMAÇÃO – CIN**

# CIN 7941 – PLANO DE ENSINO (2021.2)

## 1 IDENTIFICAÇÃO

**DISCIPLINA: CIN 7941 – Algoritmos II**
**CARGA HORÁRIA:** 36h semestrais / 2h semanais
**PROFESSOR:** Moisés Lima Dutra (*moises.dutra@ufsc.br*)
**OFERTA:** Optativa para o Curso de Graduação em Ciência da Informação e áreas afins
**HORA E LOCAL:** Oferta remota em 2021.2, com atividades síncronas e assíncronas
As atividades síncronas ocorrerão nas terças-feiras, das 10h10 às 11h50

## 2 EMENTA

Decomposição e estruturação de problemas lógicos. Desenvolvimento intermediário de algoritmos. Prática de resolução de problemas de média complexidade por meio de algoritmos.

## 3 OBJETIVOS

**3.1 Objetivo Geral**
Desenvolver a capacidade de decompor, estruturar e resolver problemas de média complexidade por meio de algoritmos.

**3.2 Objetivos Específicos**
3.2.1 Decompor problemas de média complexidade em elementos algorítmicos.
3.2.2 Estruturar esses elementos por meio da utilização técnicas e ferramentas avançadas de programação.
3.2.3 Produzir soluções algorítmicas com os elementos estruturados.

## 4 CONTEÚDO PROGRAMÁTICO

**4.1 Funções**
4.1.1 Chamadas
4.1.2 Parâmetros
4.1.3 Conversões de tipo
4.1.4 Recursividade
4.1.5 Lambda

**4.2 Coleções**
4.2.1 Listas
4.2.2 Dicionários
4.2.3 Tuplas

**4.3 Arquivos**
4.3.1 Escrita
4.3.2 Leitura
4.3.3 Percorrimento

**4.4 Bancos de Dados**
4.4.1 Criação
4.4.2 Atualização
4.4.3 Pesquisa
4.4.4 APIs

**4.5 Classes**
4.5.1 Atributos
4.5.2 Métodos
4.5.3 Instâncias

## 5 METODOLOGIA

Aulas expositivas e resolução de exercícios, baseadas na compreensão e estruturação de problemas e no desenvolvimento de algoritmos.

As atividades remotas consistirão de quatro tipos:

- Aulas expositivas (**atividade síncrona**): 1 ou 2 horas-aula
- Resolução de listas de exercícios (**atividade assíncrona**): 1 ou mais semanas
- Retiradas de dúvidas e acompanhamento do desenvolvimento das atividades (**atividade síncrona**): 1 ou 2 horas-aula
- Provas individuais (**atividade assíncrona**): 24 horas

Ambiente Virtual de Aprendizagem (principal instrumento de suporte da disciplina):

- **Moodle** (http://moodle.ufsc.br/). Deve ser **consultado todas as semanas** pelos alunos.

Ferramentas de suporte para as aulas remotas:

- Moodle (ferramenta-base)
- Google Meet (primordialmente, para as atividades síncronas)
- Chat RNP (eventualmente)
- Youtube (eventualmente)

Ferramentas on-line para o desenvolvimento das atividades práticas:

- Google Colab (sugerida, acesso on-line em: https://colab.research.google.com/)
- Jupyter Notebook (opcional, acesso on-line em: https://notebooks.inf.ufsc.br/)
- Spyder – Pacote Anaconda (opcional, necessita instalação)
- Outra (sujeita a aprovação pelo professor)

**Importante**: **Não haverá gravação de conteúdo das aulas remotas**! Pede-se aos estudantes que **observem e respeitem** esta diretiva!

## 6 AVALIAÇÃO

**Frequência:** **Cada estudante será responsável por marcar suas presenças no Moodle da disciplina** (esta opção estará disponível por 24 horas **no dia exato de cada aula**). Se programem para fazer isso!

**Conforme a Resolução do Conselho Universitário 017/Cun/97, de 30/09/97**:

- A frequência mínima obrigatória é de 75% das aulas. Enquanto vigorar o Calendário Suplementar Excepcional da UFSC, a presença física na aula será comutada pela autoanotação de presença do estudante no Moodle.

**Avaliações:**

- Serão realizadas **duas avaliações individuais assíncronas**: Prova 1 (**P1**) e Prova 2 (**P2**).
- **As questões serão disponibilizadas no Moodle no dia marcado no cronograma e cada estudante terá um prazo de 24 horas para enviar as respostas via Moodle**.

A média final (**MF**) será calculada conforme a seguinte fórmula:

$$MF = 0.4(P1) + 0.6(P2)$$

- O aluno que obtiver a **MF entre 3,0 e 5,5** e frequência suficiente poderá, ao final do semestre, realizar uma prova de recuperação (avaliação individual assíncrona) de todo o conteúdo.
- **EM CASO DE RECUPERAÇÃO**, a nota final do semesre (**NFS**) será calculada a partir da **média simples** entre a média final (**MF**) obtida durante o semestre e a nota obtida na prova de recuperação (**PR**).

$$NFS = (MF + PR) / 2$$

## 7 BIBLIOGRAFIA

Básica:


DOWNEY, Allen. **Think Python**: How to Think Like a Computer Scientist. Needham, Ma: Green Tea Press, 2012. Disponível em: <http://greenteapress.com/thinkpython/thinkpython.pdf>. Acesso em: 05 ago. 2020.

HETLAND, Magnus Lie. **Beginning Python**: From Novice to Professional. Berkeley, CA: Apress, Inc., 2005. ISBN 9781430200727 Disponível em: <http://dx.doi.org/10.1007/978-1-4302-0072-7>. Acesso em: 05 ago. 2020.

MICROSOFT DEVELOPER. **Python for Beginners**. 2020. Disponível em: https://www.youtube.com/playlist?list=PLlrxD0HtieHhS8VzuMCfQD4uJ9yne1mE6. Acesso em: 05 ago. 2020.

PYTHON.ORG. **O Tutorial Python**. 2020. Disponível em: https://docs.python.org/pt-br/3/tutorial/index.html. Acesso em: 05 ago. 2020.

PYTHON BRASIL. **Aprenda a Programar**. 2020. Disponível em: https://wiki.python.org.br/AprendaProgramar. Acesso em: 05 ago. 2020.

PYTHON BRASIL. **Tutorial Python**. 2020. Disponível em: https://wiki.python.org.br/Tutorial_Python. Acesso em: 05 ago. 2020.

SWEIGART, Al. **Automate the Boring Stuff with Python**: Practical Programming for Total Beginners. San Francisco, Ca: no Starch Press, 2015. 504 p. Disponível em: <https://automatetheboringstuff.com/>. Acesso em: 05 ago. 2020.

THE HITCHHIKER'S. **Aprendendo Python**. 2020. Disponível em: https://python-guide-pt-br.readthedocs.io/pt_BR/latest/intro/learning.html. Acesso em: 05 ago. 2020.

W3SCHOOLS. **Python Tutorial**. 2020. Disponível em: https://www.w3schools.com/python/. Acesso em: 05 ago. 2020.

Complementar:

ARAÚJO, Everton Coimbra de. **Algoritmos**: fundamento e prática. 3. ed. ampl. e atual. Florianópolis: Visual Books, 2007. 414 p. ISBN 9788575022092.

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores**: algoritmos, Pascal, C/C++ e Java. 2. ed. São Paulo: Pearson Prentice Hall, 2008. 434p. ISBN 9788576051480.

BOOCH, Grady et al. **Object-oriented analysis and design with applications**. 3rd ed. Upper Saddle River: Addison Wesley, c2007. xxiii, 691 p. ISBN 9780201895513.

BORATTI, Isaias Camilo; OLIVEIRA, Alvaro Borges de. **Introdução à programação**: algoritmos. 3. ed. Florianópolis: Visual Books, 2007. 158 p. ISBN 9788575022153.

BORGES, Luiz Eduardo. **Python**: para desenvolvedores. São Paulo: Novatec, 2014. 318 p. ISBN 9788575224052.

BLAHA, Michael; RUMBAUGH, James. **Modelagem e projetos baseados em objetos com UML 2**. 2. ed. rev. e atual. Rio de Janeiro: Elsevier, 2006. xvii, 496 p. ISBN 8535217533.

CHURCHER, Clare. Beginning SQL queries: From novice to professional. Berkeley: Apress, Inc., 2008. ISBN 9781430205500. Disponível em: http://dx.doi.org/10.1007/978-1-4302-0550-0. Acesso em: 30 ago. 2019.

DOWNEY, Allen. **Think Python**: How to Think Like a Computer Scientist. Needham, Ma: Green Tea Press, 2012. Disponível em: <http://greenteapress.com/thinkpython/thinkpython.pdf>. Acesso em: 30 ago. 2019.

GUIMARÃES, Angelo de Moura; LAGES, Newton Alberto de Castilho. **Algoritmos e estruturas de dados**. Rio de Janeiro: Livros Tecnicos e Cientificos, 1985. 216p. ISBN 8521603789.

HETLAND, Magnus Lie. **Beginning Python**: From Novice to Professional. Berkeley, CA: Apress, Inc., 2005. ISBN 9781430200727 Disponível em: <http://dx.doi.org/10.1007/978-1-4302-0072-7>. Acesso em: 30 ago. 2019.

ICODE ACADEMY. **Python Programming**: Your Step By Step Guide To Easily Learn Python in 7 Days (Python for Beginners, Python Programming for Beginners, Learn Python, Python Language). Icode Academy: Independently Published, 2017. 97 p.

LANGTANGEN, Hans Petter. **Python Scripting for Computational Science**. Third Edition. Berlin: Springer-Verlag Berlin Heidelberg, 2008. (Texts in Computational Science and Engineering, 1611-0994; 3).

LOPES, Anita; GARCIA, Guto. **Introdução à Programação:** 500 Algoritmos Resolvidos. Rio de Janeiro: Elsevier, 2002. 469 p.

MCGUGAN, Will. **Beginning Game Development with Python and Pygame**: From Novice to Professional. Berkeley: Apress, Inc., 2007. ISBN 9781430203254. Disponível em: <http://dx.doi.org/10.1007/978-1-4302-0325-4>. Acesso em: 30 ago. 2019.

MATTHES, Eric. **Python Crash Course**: A Hands-On, Project-Based Introduction to Programming. San Francisco, Ca: no Starch Press, 2015. 560 p.

MATTHEW, Neil; STONES, Richard. Beginning databases with PostgreSQL: from novice to professional. Second Edition. Berkeley, CA: Apress, Inc., 2005. Disponível em: http://dx.doi.org/10.1007/978-1-4302-0018-5. Acesso em: 30 ago. 2019.

MENEZES, Nilo Ney Coutinho. **Introdução à Programação com Python:** Algoritmos e lógica de programação para iniciantes. 2. ed. Novatec: São Paulo, 2014. 328 p.

MUELLER, John Paul. **Começando a Programar em Python para Leigos.** Rio de Janeiro: Alta Books, 2016. 379 p. (For Dummies).

ORACLE CORPORATION. MySQL 8.0 reference manual / SQL statement syntax. Pre-general availability draft: 2017-06-03. Disponível em: https://dev.mysql.com/doc/refman/8.0/en/sql- syntax.html. Acesso em: 30 ago. 2019.

ORACLE CORPORATION. MySQL documentation: MySQL reference manuals, 2014. Disponível em: http://dev.mysql.com/doc/. Acesso em: 19 dez. 2016.OWENS, Michael. The Definitive Guide to SQLite. Berkeley: Apress, Inc., 2006. ISBN 9781430201724. Disponível em: http://dx.doi.org/10.1007/978-1-4302-0172-4. Acesso em: 30 ago. 2019.

PAIVA, Severino. **Introdução à programação**: do algoritmo às linguagens atuais. Rio de Janeiro: Ciência Moderna, 2008. 192 p. ISBN 9788573936827.

PARSIAN, Mahmoud. JDBC Metadata, MySQL, and Oracle Recipes: A Problem-Solution Approach. Berkeley: Apress, Inc., 2006. ISBN 9781430201342. Disponível em: http://dx.doi.org/10.1007/978-1-4302-0134-2. Acesso em: 30 ago. 2019.

PREISS, Bruno R. **Estruturas de dados e algoritmos**: padroes de projetos orientados a objetos com java. Rio de Janeiro: Campus, c2001. xvi, 566p. ISBN 8535206930.

SCHWARTZ, Baron; ZAITSEV, Peter; TKACHENKO, Vadim. High performance mySQL. 3rd ed. Beijing: O'Reilly, 2012. xxviii, 793 p. ISBN 9781449314286.

SWEIGART, Al. **Automate the Boring Stuff with Python**: Practical Programming for Total Beginners. San Francisco, Ca: no Starch Press, 2015. 504 p. Disponível em: <https://automatetheboringstuff.com/>. Acesso em: 30 ago. 2019.

SUMMERFIELD, Mark. **Programação em Python 3:** Uma Introdução Completa à Linguagem Python. Rio de Janeiro: Alta Books, 2013. 506 p.

YARGER, Randy Jay; REESE, George; KING, Tim. MySQL and mSQL. 1st ed. Sebastopol: O'Reilly, c1999. xv, 487 p. ISBN 1565924347.

ZELLE, John. **Python Programming**: An Introduction to Computer Science. 3. ed. Portland, Or: Franklin, Beedle & Associates, 2016. 552 p.

| 8 CRONOGRAMA * | | |
|---|---|---|
| **AULA** | **DATA** | **CONTEÚDO** |
| 1 | 26/10 | Aula Expositiva: Apresentação da Disciplina (**atividade síncrona**): 30 minutos.<br>Aula Expositiva: Funções (**atividade síncrona**): 1,5 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 2 semanas. |
| 2 | 02/11 | **Feriado (Finados)** |
| 3 | 09/11 | Aula Expositiva: Funções (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 1 semana. |
| 4 | 16/11 | Aula Expositiva: Coleções (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 1 semana. |
| 5 | 23/11 | Aula Expositiva: Exceções (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 1 semana. |
| 6 | 30/11 | Aula Expositiva: Funções (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 1 semana. |
| 7 | 07/12 | Aula Expositiva: Coleções Avançadas (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 8 semanas. |
| 8 | 14/12 | Aula de Acompanhamento: Retirada de Dúvidas (**atividade síncrona**): 2 horas-aula. |
| **RECESSO ESCOLAR** | | |
| 9 | 01/02 | **(P1): Prova 1** – via Moodle (**atividade assíncrona**): 24 horas. |
| 10 | 08/02 | Aula Expositiva: Funções Avançadas (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 1 semana. |
| 11 | 15/02 | Aula Expositiva: Arquivos e Bancos de Dados (**atividade síncrona**): 1 hora-aula.<br>Lista de Exercícios (**atividade assíncrona**): 3 semanas. |
| 12 | 22/02 | Estudo e Prática Individual (**atividade assíncrona**). |
| 13 | 01/03 | **Feriado (Carnaval)** |
| 14 | 08/03 | Aula de Acompanhamento: Retirada de Dúvidas (**atividade síncrona**): 2 horas-aula. |
| 15 | 15/03 | **(P2): Prova 2** – via Moodle (**atividade assíncrona**): 24 horas. |
| 16 | 22/03 | **(PR): Prova de Recuperação** – via Moodle (**atividade assíncrona**): 24 horas. |

***Cronograma sujeito a alterações e adaptações ao longo do semestre.***

**IMPORTANTE:** Devido à excepcionalidade do semestre, poderão ser disponibilizados materiais de conteúdo e atividades via Moodle durante as semanas que possuem feriado, de maneira a serem trabalhados de maneira assíncrona. Isso também ocorrerá quando alguma eventualidade impossibilitar a ocorrência da atividade síncrona previamente agendada!