

## Pontifícia Universidade Católica do Paraná

### Plano de Ensino

|                                   |  |                      |                     |
|-----------------------------------|--|----------------------|---------------------|
| <b>Escola/Campus:</b>             | Politécnica  |                      |                     |
| <b>Cursos:</b>                    | Bacharelado em Ciência da Computação (BCC)                       | <b>Ano/Semestre:</b> | 2024/1              |
| <b>Código/Nome da disciplina:</b> | Complexidade de Algoritmos                                       |                      |                     |
| <b>Carga Horária:</b>             | 80 horas   |                      |                     |
| <b>Requisitos:</b>                | Raciocínio Algorítmico, Estrutura de Dados e Matemática Discreta |                      |                     |
| <b>Créditos:</b>                  |  | <b>Período:</b> 7    | <b>Turma:</b> U     |
| <b>Professor Responsável:</b>     |  |                      |                     |
|                                   | Edson Emilio Scalabrin   |                      | <b>Turno:</b> Noite |
|                                   |  |                      |                     |

## 1. Ementa

A disciplina, de natureza teórico-prática, é ofertada no sétimo período da Ciência da Computação. O estudante aplica conceitos e métodos da matemática discreta na análise de complexidade de algoritmos iterativos, recursivos e randomizados. Ao término da disciplina, é capaz de projetar algoritmos eficientes e expressar suas complexidades, utilizando métodos de contagem, resolução de recorrência e notação assintótica de funções de complexidade. Os estudantes trabalham em equipes para verificar os benefícios da análise de algoritmos por meio de experimentos realistas com vistas as análises de tempo de processamento e uso de memória. Para frequentar a disciplina é necessário conhecimento em raciocínio algorítmico, estrutura de dados e matemática discreta.

## 2. Relação com disciplinas precedentes e posteriores

Esta disciplina é parte integrante do eixo de fundamentos do curso. A disciplina fornece a base conceitual e tecnológica necessária para projeto de algoritmos e análise de complexidades de algoritmos.

### Disciplinas anteriores

- Raciocínio algorítmico,
- Estrutura de dados, e
- Matemática discreta

### Disciplinas posteriores

- Projeto de Conclusão do Curso

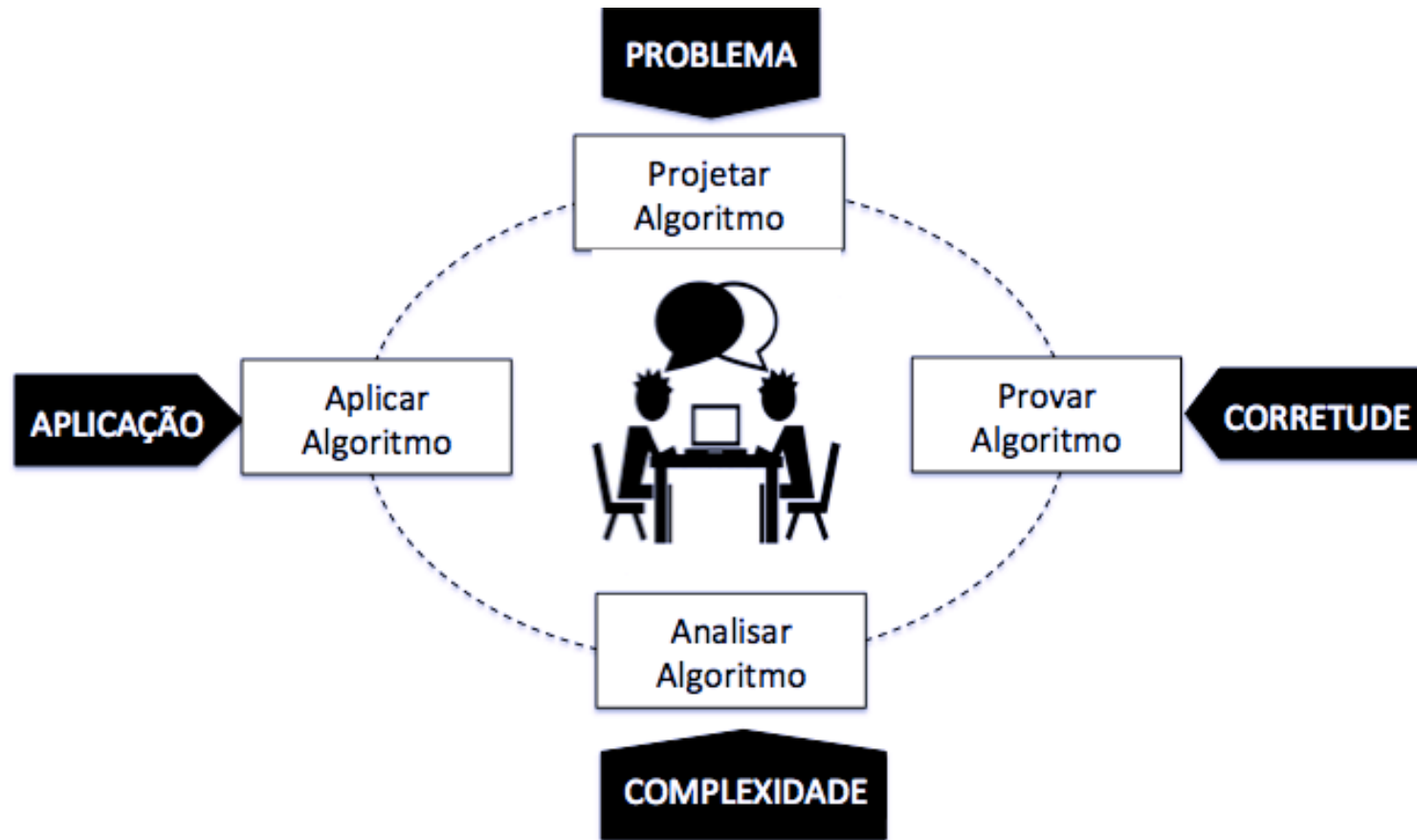
## 3. Temas de estudo

|              |  |
|--------------|--|
| Tema (TE1):  | Análise de algoritmos.                           |
| Tema (TE2):  | Indução matemática.                              |
| Tema (TE3):  | Somatório e soma telescópica.                    |
| Tema (TE4):  | Divisão e conquista.                             |
| Tema (TE5):  | Estratégia gulosa.                               |
| Tema (TE6):  | Resolução por substituição e árvore de recursão. |
| Tema (TE7):  | Teorema mestre.                                  |
| Tema (TE8):  | Algoritmos e esquemas de aproximação.            |
| Tema (TE9):  | Classes e relações de complexidade.              |
| Tema (TE10): | NP-completude.                                   |

## 4. Resultados de Aprendizagem

| Resultados de Aprendizagem  | Temas de Estudo          | Elemento de Competência (Internaliza, Mobiliza, Certifica) e Competência   |
|---|--------------------------|--|
| <b>RA1:</b> Emprega técnicas de contagem, soma e prova matemática na análise de algoritmos iterativos para problemas de computação, de acordo com o contexto de aplicação e preceitos éticos.   | TE1, TE2, TE3            | <p><b>Competência A</b> - Analisar a complexidade de algoritmos, usando técnicas de contabilização de instruções, estimativas de tempo de execução e espaço de memória, previsão de crescimento de recursos exigidos, de forma autônoma e contextualizada.</p> <p><b>(Mobiliza) Elemento de competência A1.</b> Analisar a complexidade de tempo e espaço de algoritmos iterativos, com ferramentas e métodos de prova, expressando em notação assintótica, gráfica e contextualizada.</p>   |
| <b>RA2.</b> Emprega técnicas de contagem, soma, resolução de recorrência, prova matemática na análise de algoritmos recursivos e randomizados para problemas de computação, de acordo com as características do problema a resolver e contexto. | TE1, TE4, TE5, TE6, TE7, | <p><b>Competência A</b> - Analisar a complexidade de algoritmos, usando técnicas de contabilização de instruções, estimativas de tempo de execução e espaço de memória, previsão de crescimento de recursos exigidos, de forma autônoma e contextualizada.</p> <p><b>(Mobiliza) Elemento de competência A2.</b> Analisar a complexidade de tempo e espaço de algoritmos recursivos, com estratégias de redução da complexidade e métodos de resolução de recorrência, integrando projeto de algoritmo, equação de recorrência, método de resolução e contexto.</p> |
| <b>RA3.</b> Seleciona técnicas e algoritmos para resolver de forma aproximada problemas intratáveis, de acordo com as características do problema e contexto, e preceitos éticos e legais.  | TE1, TE8, TE9, T10       | <p><b>Competência A</b> - Analisar a complexidade de algoritmos, usando técnicas de contabilização de instruções, estimativas de tempo de execução e espaço de memória, previsão de crescimento de recursos exigidos, de forma autônoma e contextualizada.</p> <p><b>(Mobiliza) Elemento de competência A3.</b> Selecionar algoritmos e esquemas de aproximação para problemas intratáveis, considerando relações entre classes de problema, aproximação algorítmica e contexto.</p>   |

## 5. Mapa Mental



## 6. Metodologia e Avaliação

Tabela de Resultados de Aprendizagem (RA), correspondentes Indicadores de Desempenho (ID), método de ensino e aprendizagem e processos de avaliação.

| Alinhamento Construtivo   |  |   |   |
|---|--|---|---|
| Resultado de aprendizagem   | Indicadores de desempenho  | Processos de Avaliação  | Métodos ou técnicas empregados**  |
| <b>RA1:</b> Emprega técnicas de contagem, soma e prova matemática na análise de algoritmos interativos para problemas de computação, de acordo com o contexto de aplicação e preceitos éticos.  | <b>ID01:</b> Contabiliza instruções de algoritmos.   | <b>Avaliação diagnóstica:</b> resolução de exercícios com cunho conceitual-teórico durante as aulas.<br><b>Feedback:</b> coletivo e imediato, durante as aulas.   | <b>Métodos empregados:</b> PBL e Peer Instruction<br><br><b>Meios de interação:</b> CANVAS, Mentimeter e Zoom. O Zoom será usado apenas em caso extraordinário. |
|   | <b>ID02:</b> Realiza prova por indução.<br><br><b>ID03:</b> Aplica telescópica para resolver somatórios.<br><br><b>ID04:</b> Aplica notação assintótica.<br><br><b>ID05:</b> Análise complexidade termos de pior caso, melhor caso e caso médio. | <b>Avaliação formativa:</b> resolução de exercícios.<br><b>Feedback:</b> coletivo em sala de aula.<br><br><b>Avaliação somativa:</b><br>a) TRABALHO de resolução de problemas, com foco em algoritmos de iterativos, feita em equipes.<br><b>Feedback:</b> coletivo, para a equipe, com questionamento individual. <b>Recuperação:</b> ocorrerá no formato de trabalhos extraclasse, de acordo com a necessidade de cada aluno.<br>b) PROVA escrita com questões relativas à análise de complexidade de algoritmos iterativos em contexto reduzido.<br><b>Feedback:</b> individual.<br><b>Recuperação:</b> ocorrerá na semana de recuperação. |   |
| <b>RA2.</b> Emprega técnicas de contagem, soma, resolução de recorrência, prova matemática na análise de algoritmos recursivos e randomizados para problemas de computação, de acordo com as características do problema a resolver e contexto. | <b>ID06:</b> Aplica estratégia divisão e conquista, estratégia gulosa.<br><br><b>ID07:</b> Aplica métodos de resolução, substituição, árvore de recursão, teorema mestre.  | <b>Avaliação diagnóstica:</b> resolução de exercícios com cunho conceitual-teórico durante as aulas.<br><b>Feedback:</b> coletivo e imediato, durante as aulas.<br><br><b>Avaliação formativa:</b> resolução de exercícios.<br><b>Feedback:</b> coletivo em sala de aula.<br><br><b>Avaliação somativa:</b><br>c) TRABALHO de resolução de problemas, com foco em algoritmos de recursivos, feita em equipes.   | <b>Métodos empregados:</b> PBL e Peer Instruction<br><br><b>Meios de interação:</b> CANVAS, Mentimeter e Zoom. O Zoom será usado apenas em caso extraordinário. |

|  |   |   |  |
|--|---|---|--|
|  |   | <p><b>Feedback:</b> coletivo, para a equipe, com questionamento individual. <b>Recuperação:</b> ocorrerá no formato de trabalhos extraclasse, de acordo com a necessidade de cada aluno.</p> <p>d) PROVA escrita com questões relativas à análise de complexidade de algoritmos recursivos em contexto reduzido.</p> <p><b>Feedback:</b> individual.</p> <p><b>Recuperação:</b> ocorrerá na semana de recuperação.</p>  |  |
| <p><b>RA3.</b> Selecciona técnicas e algoritmos para resolver de forma aproximada problemas intratáveis, de acordo com as características do problema e contexto, e preceitos éticos e legais.</p> | <p><b>ID08:</b> Mapeia classes e relações de complexidade, P, NP, NP-completo.</p> <p><b>ID09:</b> Mapeia soluções de aproximação para problemas intratáveis.</p> | <p><b>Avaliação diagnóstica:</b> resolução de exercícios com cunho conceitual-teórico durante as aulas.</p> <p><b>Feedback:</b> coletivo e imediato, durante as aulas.</p> <p><b>Avaliação formativa:</b> resolução de exercícios.</p> <p><b>Feedback:</b> coletivo em sala de aula.</p> <p><b>Avaliação somativa:</b></p> <p>a) TRABALHO de resolução de problemas, com foco em algoritmos de aproximação, feita em equipes.</p> <p><b>Feedback:</b> coletivo, para a equipe, com questionamento individual. <b>Recuperação:</b> ocorrerá no formato de trabalhos extraclasse, de acordo com a necessidade de cada aluno.</p> <p>b) PROVA escrita com questões relativas a técnicas e algoritmos de resolução de problemas por aproximação em contexto reduzido.</p> <p><b>Feedback:</b> individual.</p> <p><b>Recuperação:</b> ocorrerá na semana de recuperação.</p> | <p><b>Métodos empregados:</b> PBL e Peer Instruction</p> <p><b>Meios de interação:</b> CANVAS, Mentimeter e Zoom. O Zoom será usado apenas em caso extraordinário.</p> |

## Avaliação Somativa:

|                   | Resultados de Aprendizagem (RA) |           |           |
|-------------------|---------------------------------|-----------|-----------|
|                   | RA1                             | RA2       | RA3       |
| Item de avaliação | Do 1 ao 5                       | Do 6 ao 7 | Do 8 ao 9 |
| Trabalho          |                                 |           | 0,5       |
| Atividades        | 0,2                             | 0,2       |           |
| Prova             | 0,8                             | 0,8       | 0,5       |
| Nota do RA        | 0,30                            | 0,40      | 0,30      |

**Critério de aprovação:** nota do semestre igual ou superior a 7 (sete) e frequência mínima de **75%**.

### Recuperação Parcial:

- A recuperação parcial dos RAs 1 e 2 ocorrerão durante o semestre letivo por meio de provas específicas, conforme o cronograma.
- É elegível para realizar a recuperação parcial o estudante que obteve nota menor que 7 no RA.
- A nota máxima do estudante no respectivo RA após a recuperação parcial será igual a 7,0 (sete). Assim, os estudantes que conseguirem uma nota entre 7,0 e 10,0 na recuperação terão a nota máxima igual a 7,0 (sete).

### Recuperação Final:

- A Semana de Recuperação Estendida permitirá recuperar os RAs 1, 2 e 3.
- Somente o estudante que obteve nota semestral maior ou igual a 4 e menor que 7 poderá fazer a recuperação da Semana de Recuperação Estendida.
- Somente os RAs com nota inferior a 7 poderão ser recuperados na Semana de Recuperação Estendida.
- A média semestral para o estudante que realizar avaliações na Semana de Recuperação Estendida é limitada a 7 (sete).

## 7. Cronograma de atividades

| 2024 | Dia | RA  | Atividades pedagógicas<br>(* = atividades que vão gerar entregas para atribuir frequência)  | Em aula /<br>TDE | Carga<br>horária da<br>atividade | #  |
|------|-----|-----|---|------------------|----------------------------------|----|
| 02   | 20  | RA1 | [Introdução] Apresentação do plano de ensino, ementa, formato de avaliação. Apresentação de TDE. Contextualização sobre Complexidade de Algoritmos. | Em aula          | 4h/a                             | 1  |
|      | 27  | RA1 | [Teoria & Prática] Análise de algoritmos: análise de caso médio, pior caso e amortizada.  | Em aula          | 4h/a                             | 2  |
| 03   | 05  | RA1 | [Teoria & Prática] Análise de algoritmos: notação assintótica.  | Em aula          | 4h/a                             | 3  |
|      | 12  | RA1 | [Teoria & Prática] Indução matemática e Somatório e soma telescópica.   | Em aula          | 4h/a                             | 4  |
|      | 19  | RA1 | [Teoria & Prática] Revisão I<br>[Avaliação Somativa] Atividade I lista de exercícios  | Em aula          | 4h/a                             | 5  |
|      | 26  | RA1 | Assistir o filme “O HOMEM QUE VIU O INFINITO” e relatar os temas tratados que possuem relação a disciplina.   | TDE              | 8h/a                             | 6  |
|      |     | RA1 | [Avaliação Somativa] prova I  | Em aula          | 4h/a                             | 6  |
| 04   | 02  | RA1 | <b>Recuperação R01</b><br>[Teoria & Prática] Projeto de algoritmos: divisão e conquista, e estratégia gulosa.                                       | Em aula          | <b>2h/a</b><br>2h/a              | 7  |
|      | 09  | RA2 | [Teoria & Prática] Complexidade de tempo: substituição.   | Em aula          | 4h/a                             | 8  |
|      | 16  | RA2 | [Teoria & Prática] Complexidade de tempo: árvore de recursão. Complexidade de tempo: teorema mestre.  | Em aula          | 4h/a                             | 9  |
|      | 23  | RA2 | [Avaliação Somativa] Implementação de algoritmos e análise da complexidade.   | Em aula          | 4h/a                             | 10 |
| 05   | 07  | RA2 | [Teoria & Prática] Revisão II   | Em aula          | 4h/a                             | 11 |
|      | 14  | RA2 | [Avaliação Somativa] prova II   | Em aula          | 4h/a                             | 12 |
|      | 21  | RA3 | <b>Recuperação R02</b><br>Problemas intratáveis: algoritmos de aproximação.   | Em aula          | <b>2h/a</b><br>2h/a              | 13 |
|      | 28  | RA3 | Classes de complexidade e relações entre classes de complexidade, Np-completude.  | Em aula          | 4h/a                             | 14 |
| 06   | 04  | RA4 | [Avaliação Somativa] prova III  | Em aula          | 4h/a                             | 15 |
|      | 11  | RA4 | [Avaliação Somativa] Trabalho   | Em aula          | 4h/a                             | 16 |
|      | 16  | RA4 | <b>Recuperação R03</b>  | Em aula          | 4h/a                             | 17 |
|      | 25  | RA4 | <b>Semana Estendida de Recuperação</b>  | Em aula          | 4h/a                             | 18 |



## 8. Referências

### 8.1. Básica

Todas atividades contarão com o apoio de material preparado pelo professor sob a forma de slides, vídeos e exercícios implementados em linguagem de programação disponibilizados no ambiente virtual de aprendizagem. Os seguintes livros poderão ser consultados na biblioteca da PUCPR pelos estudantes como fonte complementar:

1. C. E. Leiserson, C. Stein, R. L. Rivest, T. H. Cormen, Algoritmos - Teoria e Prática. Editora Campus, 2002. ISBN 85-352-0926-3.
2. TOSCANI, Laura V.; VELOSO, Paulo A. S. Complexidade de algoritmos: análise, projeto e métodos. Porto Alegre: Sagra Luzzatto, 2001. 202 p.
3. ZIVIANI, Nivio. Projeto de algoritmos com implementações em Pascal e C. 5. ed. São Paulo: Pioneira, 2000. 267 p.

**Sobre a referência 1:** "Algoritmos - Teoria e Prática" é um livro abrangente escrito por C. E. Leiserson, C. Stein, R. L. Rivest e T. H. Cormen, que oferece uma análise detalhada da teoria e implementação prática de algoritmos. Com uma abordagem didática, a obra cobre uma variedade de temas, desde estruturas de dados até algoritmos avançados, fornecendo exemplos e exercícios para auxiliar na compreensão dos conceitos.

**Sobre a referência 2:** "Complexidade de algoritmos: análise, projeto e métodos" de Laura V. Toscani e Paulo A. S. Veloso é um livro que se concentra na análise e projeto de algoritmos, destacando sua complexidade. Com uma abordagem focada em métodos práticos, os autores exploram diversas técnicas para analisar e otimizar algoritmos, oferecendo insights para estudantes e profissionais da área de computação.

**Sobre a referência 3:** "Projeto de algoritmos com implementações em Pascal e C" de Nivio Ziviani é uma obra clássica que aborda o projeto de algoritmos, apresentando implementações práticas em linguagens como Pascal e C. Com uma ênfase na aplicação dos conceitos teóricos, o livro fornece uma sólida base para o desenvolvimento e análise de algoritmos, sendo uma referência essencial para estudantes e profissionais de ciência da computação.

### 8.2. Complementar

1. CORMEN, T.H., LEISERSON, C.E. e RIVEST, R.L. Introduction to Algorithms, McGraw-Hill, 1998.
2. MANBER, Udi Introduction to Algorithms: A Creative Approach.. Addison-Wesley, 1989.
3. AHO, Alfred V.; HOPCROFT, John E; ULLMAN, Jeffrey D. The design and analysis of computer algorithms. Reading: Addison-Wesley, 1974. 470 p. ISBN 0-201-00029-6
4. SIPSER, Michael. Introdução à teoria da computação. São Paulo: Thomson, 2007. 459 p. ISBN 978-85-221-0499-4 (broch.)
5. HOPCROFT, John E; ULLMAN, Jeffrey D.; MOTWANI, Rajeev. Introdução à teoria dos autômatos, linguagens e computação. Rio de Janeiro: Elsevier, 2002. 560 p. ISBN 85-352-1072-5 (broch.).

**Sobre a referência 1:** "Introdução aos Algoritmos", de Cormen, Leiserson e Rivest, é um texto fundamental no estudo de algoritmos. Ele aborda uma ampla gama de algoritmos, desde estruturas de dados até algoritmos de otimização, oferecendo uma compreensão sólida dos princípios algorítmicos. Com uma abordagem rigorosa e clara, o livro é amplamente utilizado em cursos de graduação em ciência da computação.

**Sobre a referência 2:** "Introdução aos Algoritmos: Uma Abordagem Criativa", de Manber, apresenta uma perspectiva inovadora sobre o estudo de algoritmos, incentivando a criatividade. O livro explora uma variedade de técnicas algorítmicas de forma acessível e envolvente, adequada para estudantes iniciantes e avançados. Com uma ênfase na resolução de problemas de maneira criativa, esta obra desafia os leitores a pensar de forma crítica e original sobre algoritmos.

**Sobre a referência 3:** "O Design e Análise de Algoritmos de Computador", de Aho, Hopcroft e Ullman, é um clássico na literatura de algoritmos, fornecendo uma base sólida em teoria e prática. O livro aborda os princípios fundamentais por trás da concepção e análise de algoritmos, cobrindo uma variedade de técnicas e estruturas de dados essenciais. Com exemplos claros e exercícios abrangentes, é uma leitura indispensável para estudantes de ciência da computação.

**Sobre a referência 4:** "Introdução à Teoria da Computação", de Sipser, é uma introdução acessível e abrangente ao campo da teoria da computação. O livro explora tópicos como autômatos, gramáticas formais, complexidade computacional e teoria da decidibilidade com clareza e rigor matemático. Com uma abordagem didática e exemplos elucidativos, é uma leitura essencial para quem deseja entender os fundamentos teóricos da computação.

**Sobre a referência 5:** "Introdução à Teoria dos Autômatos, Linguagens e Computação", de Hopcroft, Ullman e Motwani, é uma obra amplamente utilizada que aborda os fundamentos teóricos por trás das linguagens formais, autômatos e computação. O livro oferece uma introdução completa e acessível a esses tópicos, com uma variedade de exemplos e exercícios para reforçar o aprendizado. É uma leitura recomendada para estudantes e profissionais interessados na teoria da computação e suas aplicações.