



# Instituto de Computação - UNICAMP



MC658 - Projeto e Análise de Algoritmos III  
Professor Cid C. de Souza (turma A)

1º semestre de 2019

<a href="#">Novidades</a>	<a href="#">Docente</a>	<a href="#">Locais e horários</a>	<a href="#">Objetivos</a>	<a href="#">Programa da disciplina</a>	<a href="#">Referências bibliográficas</a>	<a href="#">Material didático</a>	<a href="#">Avaliação</a>	<a href="#">Listas de exercícios</a>	<a href="#">Datas importantes</a>
---------------------------	-------------------------	-----------------------------------	---------------------------	--	--	-----------------------------------	---------------------------	--------------------------------------	-----------------------------------

## ● Novidades: Consulte esta seção frequentemente.

1. Início das aulas.  
[28/02/2019]

## ● Docente:

[Cid Carvalho de Souza](#)

Sala: 8 (Prédio IC1)

Email para contato: cid[at]ic[dot]unicamp[dot]br

**PED:** Natanael Ramos (naelr8[at]gmail[dot]com)

## ● Dias e locais das aulas e do atendimento:

- **aulas:** as aulas serão das 16:00 às 18:00 horas nas 3<sup>as</sup> (sala ?B??) e 5<sup>as</sup> (sala ?B??).
- **atendimentos do docente:** às terças depois da aula ou em horário previamente combinado (solicitado por *email* enviado com pelo menos 72 horas de antecedência).  
*O docente não dará atendimento em semana de prova ou exame.*
- **atendimento do PED:** início a partir da **terceira** semana de aula em horário a ser definido.  
**O atendimento do PED será suspenso se: (i) nenhum aluno comparecer nos primeiros 15 minutos; ou (b) passados os 15 minutos iniciais, a fila de espera por atendimento tiver se esgotado; ou (c) o horário de término do atendimento tiver sido atingido.**

## ● Objetivos da Disciplina:

O objetivo desta disciplina é complementar a formação dos alunos na área de algoritmos, introduzindo técnicas para lidar com problemas NP-difíceis, resolvendo-os de forma exata, heurística ou aproximada. Espera-se que ao final do semestre os alunos tenham adquirido sólidos conhecimentos sobre as técnicas e sejam capazes de compreender as vantagens e as limitações de uso de cada uma delas.

## ● Programa: (em verde encontra-se o material já coberto)

**Reduções e Classes de Problemas:** revisão do conceito de reduções entre problemas e das definições das classes P, NP, NP-completo e NP-difícil. **Algoritmos exatos para problemas NP-difíceis:** algoritmos pseudo-polinomiais; algoritmos de backtracking; algoritmos de branch-and-bound; Programação Linear Inteira e Programação por Restrições como ferramentas para resolução de problemas NP-difíceis; **Algoritmos heurísticos para problemas NP-difíceis:** definições básicas; algoritmos construtivos e algoritmos de busca local; algoritmos construtivos gulosos; algoritmos de busca local; meta-heurísticas; **Algoritmos aproximados** definições básicas: aproximação absoluta; exemplos de aproximação absoluta; inaproximabilidade em aproximação absoluta; fator de aproximação (aproximação relativa); exemplos de fator de aproximação; inaproximabilidade em fator de aproximação; uso de PL no desenvolvimento de algoritmos aproximados; esquemas de aproximação polinomial (definição e exemplo);

## ● Referências bibliográficas:

A seguir encontra-se a bibliografia de base da disciplina com alguns comentários adicionados pelo docente visando auxiliá-lo na escolha das obras a serem pesquisadas durante os seus estudos. Note que, além desses livros, existem nas bibliotecas da UNICAMP outras excelentes obras sobre os assuntos que serão vistos em sala de aula.

1. T.H. Cormen, C.E. Leiserson, R.L.Rivest e C. Stein. Introduction to Algorithms. 2nd Edition, McGraw-Hill, 2001.  
*Comentário:* Livro básico das disciplinas de algoritmos do IC.
2. U. Manber. Introduction to Algorithms: A Creative Approach. Addison-Wesley. 1989.  
*Comentário:* Outro livro básico das disciplinas de algoritmos do IC. O capítulo 10 trata exclusivamente sobre reduções entre problemas e bastante rico em exemplos. O mesmo pode ser dito a respeito do capítulo 11 sobre NP-completude.
3. C. H. Papadimitriou e K. Steiglitz. Combinatorial Optimization: Algorithms and Complexity. Prentice-Hall, Inc.,1982.  
*Comentário:* Uma boa fonte de referência na parte de NP-completude, principalmente para explicar as técnicas de *branch-and-bound* e apresentar os conceitos básicos de algoritmos aproximados.
4. E. Horowitz e S. Sahni. Fundamentals of Computer Algorithms. Computer Science Press, 1978.  
*Comentário:* Outra boa fonte de referência para NP-completude, algoritmos aproximados, algoritmos de *branch-and-bound* e *backtracking*.
5. M. Garey e D. Johnson. Computers and Intractability: a Guide to the Theory of NP-Completeness. Freeman. 1979.  
*Comentário:* Uma espécie de referência "bíblica" sobre a Teoria da Complexidade !
6. P. J. de Rezende e J. Stolfi. Fundamentos de geometria computacional. Universidade Federal de Pernambuco, Departamento de Informática, 1994. IX Escola de Computação, Recife, 24 a 31 de julho de 1994.  
*Comentário:* Uma boa leitura sobre reduções entre problemas.
7. M.C. Goldberg e H.P.L. Luna. Otimização Combinatória e Programação Linear: modelos e algoritmos. Editora Campus. 2000.  
*Comentário:* Uma boa fonte de consulta em português sobre formulação de problemas usando PL e PLI.
8. M. Bazaraa, J. Jarvis e H. Sherali. Linear Programming and Network Flows. 2ª edição, John Wiley and Sons. 1990.  
*Comentário:* Uma boa fonte de consulta sobre PL e PLI.
9. L. Wolsey. Integer Programming. Wiley-Interscience. 1998.  
*Comentário:* Uma boa fonte de consulta sobre PLI.
10. M. H. Carvalho, M. R. Cerioli, R. Dahab, P. Feofiloff, C. G. Fernandes, C. E. Ferreira, K. S. Guimarães, F. K. Miyazawa, J. C. Pina Jr., J. Soares, Y. Wakabayashi.  
*Uma Introdução Sucinta a Algoritmos de Aproximação*, 23º Colóquio Brasileiro de Matemática, 2001 ([versão PDF](#), disponibilizada na página do [Prof. Flávio Miyazawa](#)).  
*Comentário:* Uma boa fonte de consulta sobre algoritmos aproximados.
11. M. Sipser. Introduction to the Theory of Computation. PWS Publishing Company, 1997.  
*Comentário:* Uma excelente obra sobre Teoria da Computação onde tópicos cobertos nesta disciplina são vistos com um maior formalismo.
12. F.K. Miyazawa e C.C. de Souza, *Introdução à Otimização Combinatória*, Jornadas de Atualização em Informática - Congresso da Sociedade Brasileira de Computação - JAI-SBC, 2015. ([versão pdf](#); ver capítulo 3).  
*Comentário:* uma obra recente em português que cobre bastante do conteúdo que é visto em MC658.

## ● Material didático: Transparências usadas em aula serão disponibilizadas nesta seção, *sempre que possível, antes* da data de seu uso efetivo.

## ● Avaliação:

A avaliação será baseada nas notas de duas provas (P1 e P2), nas notas de trabalhos práticos (TPs) que serão passados ao longo do semestre.

Nenhuma destas atividades valendo nota será antecipada por solicitação de um aluno ou terá substitutiva. **Sem exceções.**

A partir destas avaliações, as notas do semestre e final serão calculadas da seguinte forma:

- **Média ponderada dos trabalhos práticos:**  $MT = (W1 \times T1 + W2 \times T2 + \dots + Wn \times Tn) / (W1 + W2 + \dots + Wn)$ , onde  $3 \leq n \leq 5$  e  $W_i$  é o peso do trabalho  $i$  podendo ser 1, 2 ou 3 dependendo do grau de dificuldade do trabalho.
- **Média geométrica das provas:**  $MP = \sqrt{P1 \times P2}$
- **Média do semestre:** (\* antes do exame \*)  
se  $MP \geq 5,0$  e  $MT \geq 5,0$  então  $MS = (7 \times MP + 3 \times MT) / 10$   
se não  $MS = \min \{ 4,9, (7 \times MP + 3 \times MT) / 10 \}$   
*Observações: se  $MS < 2,5$  ou  $MS \geq 5,0$ , o aluno estará impedido de fazer o exame.*
- **Média final:** (\* após o exame \*)  
se o aluno fez o exame então  
■  $MF = \min \{ 5,0, (MS + E) / 2 \}$   
se não (\* aluno não fez exame \*)  
■ se  $MS < 2,5$  ou  $MS \geq 5,0$  então  $MF = MS$  (\* aluno impedido de fazer o exame \*)  
■ se não  $MF = MS / 2$  (\* aluno deveria ter feito o exame, mas não fez \*)
- **Resultado final:**  
se o aluno não teve a frequência mínima então ele **REPROVOU-SE** por falta se não  
■ se  $MF \geq 5,0$  então o aluno **APROVOU-SE**  
■ se não o aluno **REPROVOU-SE** por nota

## Sobre os trabalhos práticos:

Alguns trabalhos práticos terão de ser feitos individualmente enquanto outros deverão ser feitos em grupos de 2 (excepcionalmente 3) alunos. Neles serão propostos problemas NP-difíceis que deverão ser tratados com uso das técnicas vistas na disciplina. A entrega dos trabalhos consistirá de código e de um relatório de análise de resultados em formato a ser divulgado oportunamente. O prazo para execução dos trabalhos poderá variar entre uma e três semanas, dependendo do grau de dificuldade dos mesmos.

#### Observações:

1. Não haverá provas ou trabalhos substitutivos.
2. Todas as provas realizadas durante o semestre, bem como o exame final, serão sem consulta.
3. Qualquer tentativa de fraude nas provas, no exame ou nos trabalhos práticos implicará em média final (MF) igual a ZERO para todos os envolvidos, sem prejuízo de outras sanções.
4. Qualquer pedido de **revisão da nota** atribuída a uma avaliação realizada durante o semestre deverá ser feito em no **máximo 2 dias** (úteis) contados a partir da data de divulgação dos resultados daquela avaliação. No caso do **exame**, este prazo será de **24 horas**. Solicitações que não atendam a esta exigência serão automaticamente negadas pelo docente.

#### • Listas de exercícios:

Ao longo do semestre serão propostas listas de exercícios que deverão ser feitas para uma maior fixação dos tópicos apresentados em classe. O conteúdo dos exercícios é considerado parte integrante do material visto e tratado como parte coberta da matéria. A entrega deles não será cobrada. Entretanto, para o bom aprendizado da disciplina é necessário que cada aluno tente fazer os exercícios **individualmente** e só depois discutí-los em grupo. Dúvidas ou dificuldades devem ser discutidas com o docente ou com o PED.

#### • Datas Importantes:

- [Calendário oficial da DAC](#). Visite esta página para saber quais as datas de alteração de matrícula, desistência de disciplinas e dos períodos sem atividade.
- 28/02 (qui): início das aulas.
- 16/04 (ter): **primeira prova (P1)**.
- 21/05 (ter): **reunião de avaliação de cursos. Só não haverá aula se houver coincidência de horário.**
- 25/06 (ter): **segunda prova (P2)**.
- 11/07 (qui): **exame (E)**.

*Observação:* as datas de divulgação do enunciado, disponibilização de dados de entrada e de entrega dos códigos e relatórios referentes aos trabalhos práticos serão divulgadas oportunamente e com a antecedência devida no correr do semestre.