

MC202 - Estruturas de Dados

Prof. Alexandre Xavier Falcão (afalcao@ic.unicamp.br)
PEDS: Júlia Silva (juliaborgescs@gmail.com) e Azael Sousa (azaelsousa@gmail.com)
PAD: Ulisses Malta Santos (ra140958@students.ic.unicamp.br)

Segundo semestre de 2016
Turmas G e H
www.ic.unicamp.br/~afalcao/mc202

Objetivos

Este curso abordará diferentes formas de organizar e manipular dados (informações) na memória primária, como via de regra, mas também como acessar rapidamente dados que estão na memória secundária do computador. Cada forma de organização é denominada uma estrutura de dados, as quais são especificadas em conjunto com os algoritmos de manipulação dos dados. O curso exercitará a noção de complexidade de algoritmo para fins de escolha da estrutura de dados mais adequada a um dado problema e a implementação eficiente de seus algoritmos.

Horários e atendimento

O atendimento pelo professor será prestado logo após as aulas teóricas e o atendimento pelos auxiliares didáticos será prestado durante as seções de laboratório. Se houver necessidade de atendimento em outros horários, entre em contato com o professor, ou com os auxiliares didáticos, via e-mail.

Turmas	Dia	Sala	Hora
G, H	3	CB02	16:00-18:00
G, H	5	CB02	16:00-18:00
G, H	4	SI10/SI05	16:00-18:00

Tabela 1: Aulas teóricas (terças e quintas) e práticas (quartas).

Listas de Exercícios

Ao longo do período serão indicados vários exercícios, os quais não precisarão ser entregues, mas poderão cair nas provas.

Laboratórios

Haverá n projetos de laboratório durante o semestre, com $6 \leq n \leq 8$. Os projetos deverão ser feitos em linguagem C e submetidos no sistema <https://run.codes>. Cada projeto deverá ser entregue em um prazo máximo de 7 a 10 dias devidamente especificado no enunciado.

Sejam L_i as notas dos projetos. A **média dos laboratórios** será dada por:

$$\frac{L_1 + L_2 + \dots + L_n}{n}.$$

Provas Teóricas e Exame

Haverá 5 (cinco) mini-provas teóricas Q_1, Q_2, Q_3, Q_4, Q_5 consistindo de uma questão. Cada uma dessas mini-provas terá duração de 30 minutos e será sempre feita na segunda metade da aula. **Não haverá aviso prévio sobre as datas das mini-provas** – o docente poderá aplicar uma mini-prova em qualquer dia de aula. A partir das notas dessas mini-provas calcula-se

$$P_1 := \frac{(Q_1 + Q_2 + Q_3 + Q_4 + Q_5) - \min\{Q_1, Q_2, Q_3, Q_4, Q_5\}}{4},$$

ou seja, descarta-se a menor dessas notas e calcula-se a média aritmética das notas restantes.

Ao final do semestre haverá uma prova teórica P_2 com duração de 1 hora e 40 minutos que abrangerá **toda a matéria** vista ao longo do semestre.

A **média da provas** será dada por:

$$P := \frac{P_1 + P_2}{2}.$$

Turmas	Prova	Sala	Dia
G, H	P_2	CB02	06/12/2016
G, H	Exame	CB02	22/12/2016

Tabela 2: Datas da P_2 e do exame.

Observações.

1. Não haverá provas substitutivas.
2. Não haverá projetos substitutivos.
3. Qualquer tipo de **fraude** em provas ou projetos acarretará em **média final igual a zero** para todos os envolvidos, sem prejuízo de outras sanções regimentais.

Outras datas importantes

- Avaliação e discussão de cursos: 11/10/2016.
- Feriados: 15/11/2016.
- Congresso do PIBIC: 20/10/2016.

Critério de aprovação

O aproveitamento do semestre é dado pela **média harmônica** de P e L , ou seja, por:

$$A := \begin{cases} \frac{2PL}{P+L} & \text{se } P + L > 0 \\ 0 & \text{caso contrário.} \end{cases}$$

Se $A < 5$ e $A \geq 2.5$ então o aluno poderá fazer o Exame. Seja E a nota do Exame. A **média final** do curso é dada por:

$$M := \begin{cases} A & \text{se } A \geq 5 \text{ ou } A < 2.5, \\ \min \left\{ 5, \frac{A+E}{2} \right\} & \text{se } A < 5 \text{ e } A \geq 2.5. \end{cases}$$

Note que um aluno só tem direito a fazer exame se $A \geq 2.5$. Além disso, se o aluno tiver que fazer o Exame, sua média final M será no máximo 5. O aluno está aprovado se $M \geq 5$, senão está reprovado.

Programa da disciplina

1. Estruturas ligadas: nó, apontador, variável apontadora, alocação dinâmica de memória
2. Listas ligadas simples: operações básicas
3. Comparação de listas ligadas com vetores
4. Algoritmos gerais para listas simples: enumeração, inversão, cópia, concatenação
5. Pilhas, filas, e aplicações (eliminação de recursão)
6. Intercalação (merge) de listas e mergesort; análise informal
7. Variações: listas circulares, duplamente ligadas, com cabeça.
8. Algoritmos de ordenação
9. Árvores binárias: representação e percurso (recursivo)
10. Aplicação: árvores de busca (com inserção e remoção)
11. Árvores binárias de busca balanceadas
12. Fila de prioridade (heap) implementação com vetor e heapsort
13. Árvores gerais: definição, representação por listas, percursos
14. Listas generalizadas e uso para representar estruturas ligadas em geral
15. Árvores B e generalizações
16. Introdução ao espalhamento (hashing): conceito, implementação com listas ligadas. Técnicas de espalhamento para arquivos
17. Grafos: conceito, representação por matrizes e listas ligadas
18. Percurso de grafos em largura e profundidade
19. Implementação de estruturas de dados em disco

Referências

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and Clifford Stein. Introduction to Algorithms, second edition. MIT Press, 2009.
2. A. V. Aho, J. E. Hopcroft, and J. Ullman. Data Structures and Algorithms. Addison-Wesley, 1983.
3. A. Drozdek. Estrutura de Dados e Algoritmos em C++. Thomson, 2002.
4. J. L. Szwarcfiter e L. Markenzon. Estruturas de Dados e seus Algoritmos. LTC Editora, 1994.
5. C. L. Lucchesi e T. Kowaltowski. Estruturas de Dados e Técnicas de Programação. Instituto de Computação, UNICAMP, 2004.
6. P. Feofiloff. Algoritmos em Linguagem C. Elsevier Editora Ltda., 2009.
7. G. H. Gonnet. Handbook of Algorithms and Data Structures. Addison-Wesley, 1984.
8. E. Horowitz and S. Sahni. Fundamentals of Data Structures in Pascal. Computer Science Press, 1984.
9. D. E. Knuth. The Art of Computer Programming, volume I: Fundamental Algorithms. Addison-Wesley, 1978.
10. E. M. Reingold and W. J. Hanson. Data Structures. Little Brown and Company, 1983.
11. R. Sedgewick. Algorithms in C. Addison-Wesley, 1990.
12. D. F. Stubbs and N. W. Webre. Data Structures with Abstract Data Types and Pascal. Brooks/Cole, 1985.
13. A. M. Tenenbaum, Y. Langsam, and M. J. Augenstein. Data Structures using C. Prentice-Hall, 1990.
14. N. Wirth. Algorithms + Data Structures = Programs. Prentice-Hall, 1976.
15. N. Ziviani. Projeto de Algoritmos (2a. ed.). Thomson, 2004.