

MC202 — ESTRUTURAS DE DADOS — TURMAS A, B E C
PROFESSOR: ALEXANDRE XAVIER FALCÃO
E-MAIL: afalcao@ic.unicamp.br
PRIMEIRO SEMESTRE DE 2024

Aulas As aulas teóricas serão apresentadas todas as terças na sala PB 16 (prédio básico) das 10h às 11:40h e todas as sextas na sala PB 13 (prédio básico) das 10h às 11:40h, com os 20 minutos restantes deixados para atendimento. As notas de aula e códigos usados durante as aulas estão disponibilizados na página do curso.

As primeiras aulas consistirão de um curso rápido da Linguagem C, visto que todos os alunos cursaram Algoritmos e Programação de Computadores em Python.

Os laboratórios serão apresentados e acompanhados pelo PED todas as sextas das 14h às 16h nas salas SI 08, 10, e 05 do prédio básico.

Atendimento Além dos 20 minutos de atendimento em sala de aula, o atendimento também será feito pelos monitores da disciplina em local e horários a serem divulgados no início do semestre.

Programa da Disciplina • Estruturas de dados abstratas, apontadores, e alocação de memória. • Recursão: ordenação, busca binária, e backtracking. • Listas ligadas: simples, dupla, variações, e aplicações. • Pilhas, filas, e aplicações. • Árvores binárias. • Árvores binárias de busca. • Árvores binárias de busca balanceadas. • Fila de prioridade (*heap* binário). • Espalhamento. • Grafos: representação, percursos, e algoritmos. • Árvores B.

Linguagens de Programação A linguagem de programação C será utilizada.

Tarefas de laboratório Teremos diversas tarefas de laboratório a serem entregues durante o semestre. Todas as tarefas terão um prazo total de 7 dias ou mais para a primeira entrega e deverão ser feitas **individualmente**.

Para a correção das tarefas, será utilizado o sistema Susy <https://www.ic.unicamp.br/~susy/>. Cada programa desenvolvido pelo aluno para uma tarefa específica será automaticamente avaliado por este sistema com uma nota em $[0, 10]$. As informações de como usar o sistema serão dadas no começo do semestre pelos monitores.

A nota da tarefa será proporcional ao número de casos de testes fechados resolvidos. De qualquer forma, a nota pode sofrer descontos de acordo com a qualidade do programa apresentado ou caso o programa submetido não satisfaça os critérios estabelecidos no enunciado da tarefa. Assim, mesmo que o código seja capaz de resolver todos os casos de teste da tarefa, a nota final ainda pode ser menor do que 10,0.

Cada tarefa terá uma data para a primeira entrega da solução. Porém, até o final do semestre, os alunos poderão submeter tarefas que não tenham sido submetidas dentro do prazo da primeira entrega ou resubmeter tarefas já entregues com o objetivo de aumentar a nota obtida até no máximo a nota 7,0. Caso o aluno falhe em aumentar a nota, ele continuará com a nota original, isto é, a nota não será diminuída da primeira entrega

para a entrega final. Cada tarefa será corrigida uma vez após o prazo inicial e apenas mais uma vez até o final do semestre.

Avaliação

- Cada tarefa de laboratório terá nota em $[0, 10]$ e um peso entre 1 e 5 conforme a sua dificuldade.
- A média $L \in [0, 10]$ das tarefas de laboratório será calculada pela média ponderada das notas nessas tarefas.
- Os alunos também serão avaliados por 12 questões teóricas Q_i , $i = 1, 2, \dots, 12$, valendo 1,11 cada, que serão apresentadas durante as aulas sem aviso prévio. No entanto, como o aluno só precisará estar presente em 75% das aulas, as 9 questões de maior nota serão consideradas para fins de cômputo da nota teórica T . A nota T será a soma das notas dessas 9 questões.
- A média M , antes do exame, será a média harmônica ponderada entre T (com peso 7) e L (com peso 3), isto é,

$$M = \begin{cases} 0, & \text{se } T = 0 \text{ ou } L = 0, \\ \frac{10}{\frac{7}{T} + \frac{3}{L}}, & \text{caso contrário} \end{cases} \quad (1)$$

Alunos com menos que 75% de presença nas aulas estarão reprovados por falta. Caso contrário, alunos com média $M < 5,0$ estarão em exame e serão aprovados com nota 5,0 se a média aritmética entre M e a nota do exame for pelo menos 5,0. Os demais alunos estarão aprovados por média com nota $M \geq 5,0$.

Data do exame e observações adicionais O exame final ocorrerá dia **12/07/2024**.

- Qualquer tentativa de fraude nas tarefas implicará em nota final 0 (zero) para todos os envolvidos, sem demais implicações. Exemplos de fraudes são:
 - Compartilhar trechos de código de qualquer forma.
 - Utilizar trechos de códigos da internet ou de outras fontes.
 - Copiar ou comprar uma tarefa.

Referências O professor não seguirá um livro texto específico, entretanto, os livros abaixo cobrem o que será visto em aula. Em particular, as principais referências são os livros 1 e 2 da seguinte lista.

1. R. Sedgewick, Algorithms in C. Addison-Wesley, 1990.
2. T. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Algoritmos — Teoria e Prática. Campus, 2002.

3. A. V. Aho, J. E. Hopcroft, J. Ullmann. Data Structures and Algorithms. Addison-Wesley, 1983.
4. W. Celes, R. Cerqueira, J. L. Rangel. Introdução a Estruturas de Dados. Campus, 2004.
5. M. J. Folk e B. Zoellick. File Structures. Addison-Wesley, 1992.
6. F. Lorenzi, P. N. de Mattos, T. P. de Carvalho. Estruturas de Dados. Thomson, 2007.
7. S. L. Pereira. Estruturas de Dados Fundamentais. Érica, 1996.
8. E. M. Reingold e W. J. Hanson, Data Structures. Little-Brown (1983).
9. J. L. Szwarcfiter e L. Markenzon. Estruturas de Dados e Seus Algoritmos. Editora LTC (1994).
10. D. E. Knuth, The Art of Computer Programming, Vol I: Fundamental Algorithms. Addison-Wesley (1978).
11. N. Wirth, Algorithms + Data Structures = Programs. Prentice-Hall (1976).
12. A. M. Tenenbaum. Estruturas de Dados Usando C. Makron Books, 1995.
13. N. Ziviani. Projeto de Algoritmos. Thomson, 2004.