

Atendimento O horário de atendimento será prestado sempre depois das aulas pelo professor e nas aulas presenciais de laboratório.

Os monitores farão também o atendimento por texto, áudio e vídeo através do Discord (<https://discord.com/>) em horários adicionais a serem divulgados no começo do semestre. O professor fornecerá o convite para a plataforma no começo do semestre.

Programa da Disciplina • Estruturas ligadas: nó, apontador, variável apontadora, alocação dinâmica de memória • Listas ligadas simples: operações básicas • Comparação de listas ligadas com vetores • Algoritmos gerais para listas simples: enumeração, inversão, cópia, concatenação • Pilhas, filas, e aplicações • Intercalação (merge) de listas e mergesort; análise informal • Variações: listas circulares, duplamente ligadas, com cabeça. Lista livre • Algoritmos de ordenação • Árvores binárias: representação e percurso (recursivo) • Aplicação: árvores de busca (com inserção e remoção) • Árvores binárias de busca balanceadas • Fila de prioridade (heap) implementação com vetor e heapsort • Árvores B e generalizações • Introdução ao espalhamento (hashing): conceito, implementação com listas ligadas. • Grafos: conceito, representação por matrizes e listas ligadas • Percurso de grafos em largura e profundidade

Linguagens de Programação A linguagem de programação *C* será utilizada.

Laboratórios Haverá diversos laboratórios a serem entregues durante o semestre. Todos os laboratórios terão um prazo total de 7 dias ou mais para a primeira entrega e deverão ser feitos **individualmente**.

Para a correção dos laboratórios, será utilizado o sistema de submissão <https://codepost.io>. Cada programa desenvolvido pelo aluno para um laboratório específico será automaticamente avaliado por este sistema em vários testes.

A nota do laboratório será proporcional ao número de casos de teste resolvidos. Porém, a nota pode sofrer descontos de acordo com a qualidade do programa apresentado ou caso o programa submetido não satisfaça os critérios estabelecidos no seu enunciado. Assim, mesmo que o código seja capaz de resolver todos os casos de teste fechados, a nota final ainda pode ser menor do que 10, podendo inclusive ser zerada.

Cada laboratório terá uma data limite para a entrega da solução. Essa entrega é chamada de **primeira chance**. Porém, até o final do semestre (05/12/2022), os alunos terão uma **segunda chance** para entregar o laboratório ou aumentar a nota do laboratório entregue. Nesse caso a nota adicional obtida terá um desconto de 25%. Exemplos:

- Um aluno que não entregou o laboratório na primeira chance, ao tirar nota máxima no laboratório entregue até o final do semestre terá nota 7,5.
- Um aluno que já tenha entregue o laboratório na primeira obtendo nota 6 poderá entregar o laboratório novamente, mas poderá obter nota final no máximo 9 (75% dos pontos restantes).

Caso o aluno falhe em aumentar a nota, ele continuará com a nota original, isto é, a nota não será diminuída da primeira chance para a segunda chance. Cada laboratório será corrigido uma vez após o prazo inicial e apenas mais uma vez até o final do semestre.

Formalmente, cada laboratório i terá uma nota para ℓ_i^1 atribuída para a primeira chance (sendo zero caso não for entregue) e uma nota ℓ_i^2 para a solução entregue na segunda chance (novamente sendo zero caso não for entregue). A nota ℓ_i do laboratório i será, portanto,

$$\ell_i = \ell_i^1 + \max \left\{ 0, \frac{3(\ell_i^2 - \ell_i^1)}{4} \right\}.$$

Cada laboratório i terá um peso P_i^ℓ dependendo da dificuldade. A média ML dos laboratórios será calculada como a média ponderada dos laboratórios, isto é,

$$ML = \frac{\sum_i P_i^\ell \ell_i}{\sum_i P_i^\ell}$$

Testes Durante o semestre, vários testes serão propostos na página da disciplina no Google Sala de Aula. Tais testes terão um prazo máximo para serem cumpridos pelos alunos. A correção dos testes será automaticamente feita pelo Google Sala de Aula.

Cada teste i terá um peso P_i^t dependendo da dificuldade. A média MT dos testes será calculada como a média ponderada das notas t_i dos laboratórios, isto é,

$$MT = \frac{\sum_i P_i^t t_i}{\sum_i P_i^t}$$

Avaliação

- A média M , antes do exame, será a média harmônica ponderada entre ML (com peso 8) e MT (com peso 2), isto é,

$$M = \begin{cases} 0, & \text{se } ML = 0 \text{ ou } MT = 0, \\ \frac{10}{\frac{2}{MT} + \frac{8}{ML}}, & \text{caso contrário} \end{cases} \quad (1)$$

- Todos os alunos podem fazer o exame final da disciplina. Alunos com média $M \geq 5,0$ ficam dispensados do exame final (mas podem fazê-lo, se assim desejarem).
- O exame final consiste em entregar laboratórios e testes propostos durante o semestre até 19/12/2022.
 - Laboratórios que foram entregues durante o semestre na segunda chance **não podem** ser entregues novamente.
 - Laboratórios que nunca foram entregues durante o semestre ou que foram entregues apenas na primeira chance **podem** ser entregues novamente.
 - Testes podem ser refeitos se já foram entregues durante o semestre.
 - Testes não entregues podem ser feitos durante o exame.

- A média E do exame será calculada usando a mesma fórmula utilizada para M . Nesse caso:
 - A média ML será calculada considerando também os laboratórios entregues no exame final como se tivessem sido entregues na segunda chance.
 - A média MT será calculada considerando também os testes entregues no exame final. Caso um teste seja refeito, será considerada a maior nota entre a entrega no semestre e a entrega durante o exame.
- A média final F do aluno é $F = E$, caso o aluno tenha entregue laboratórios e testes durante o exame e $F = M$, caso contrário. Note que $E \geq M$ caso o aluno faça o exame e que a nota final do aluno considera tanto o exame (caso seja feito) quanto a nota obtida durante o semestre.
- O aluno estará aprovado caso sua nota final F seja maior ou igual a 5,0 e estará reprovado caso contrário.

Fraudes Qualquer tentativa de fraude nos testes ou laboratórios implicará em nota final $F = 0$ (zero) para todos os envolvidos, sem prejuízo de outras sanções. Exemplos de fraudes são:

- Enviar trechos de códigos de qualquer forma para outros alunos.
- Receber trechos de códigos de qualquer forma de outros alunos.
- Utilizar trechos de códigos da internet ou de outras fontes sem prévia autorização do professor.
- Copiar ou comprar um laboratório.
- Disponibilizar soluções de laboratórios online antes do término completo do semestre letivo (23/12/2022).

Caso o aluno realize uma fraude e se arrependa, ele deve entrar em contato imediatamente com o professor explicando o que ocorreu e quem foram os envolvidos.

- Nesse caso, a penalidade será obter nota zero nas atividades envolvidas na fraude.
- Tal atitude só será válida se ocorrer antes do professor detectar e acusar a fraude.
- O aluno não ficará imune a ser reprovado com nota final zero por outras fraudes existentes, apenas pela fraude declarada.
- Outros participantes da fraude que não se manifestarem serão enquadrados pela regra da nota final zero descrita anteriormente.

Os alunos podem, a qualquer momento, contatar o professor, inclusive de maneira anônima, para esclarecer se determinado comportamento é considerado fraude ou não.

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.