

Estruturas de Dados

Apresentação da disciplina

1º semestre de 2020
Prof. José Martins Jr

Objetivos da disciplina

- Introdução ao estudo dos conceitos fundamentais sobre estruturas de dados
- Aplicar linguagem C na representação e manipulação das principais estruturas de dado
- Aplicar as alocações estática e dinâmica de memória
- Programar algoritmos para operar dados na memória
 - Organização
 - Manipulação
 - Busca
- Analisar a eficiência de algoritmos estudados

Organização do curso

- Aulas
 - Teóricas: conceitos e demonstrações
 - Práticas: programação
- Avaliações – **SEGUEM O PADRÃO DA FACAMP**
 - Testes (**30%** da média)
 - Prova bimestral (**30%** da média) *
 - Prova semestral (**40%** da média) *

* Agendadas pela Coordenação e divulgadas pela Secretaria

Aulas teóricas

1. Representação de arranjos de dados: vetores e matrizes.
2. Tipos de dados compostos: estruturas; alocação estática; alocação dinâmica; ponteiros.
3. Listas simplesmente encadeadas.
4. Pilhas e filas.
5. Listas duplamente encadeadas.
6. Listas circulares, nó cabeça e multilistas.
7. Análise de eficiência (assintótica) de algoritmos.
8. Ordenação de dados por, inserção, permuta e intercalação de listas.
9. Árvores: tipos e aplicações; Árvore Binária de Busca (ABB).
10. Ordenação de dados inspirada em ABB.
11. Hashing.

Aulas práticas

1. Criação e manipulação de cadeias de caracteres, operações de entrada e saída em linguagem C.
2. Operação com vetores, matrizes e arranjos circulares em linguagem C.
3. Definição, alocação (estática/dinâmica) e atribuição de dados em estruturas; acesso a dados através de ponteiros em linguagem C.
4. Operações básicas com listas simplesmente encadeadas e implementação de algoritmos para manipulação de listas em linguagem C.
5. Implementações estática e dinâmica de pilhas e de filas em linguagem C.
6. Operações com listas duplamente encadeadas em linguagem C.
7. Listas circulares, com cabeça e livres, e operações com multilistas em linguagem C.
8. Ordenação e busca utilizando os algoritmos (em linguagem C): inserção direta, bubble, shellsort, quicksort e mergesort.
9. Implementação de ABB em linguagem C e operações de inserção, percurso, busca e remoção de nós.
10. Ordenação e busca utilizando heapsort, em linguagem C.
11. Implementação de hashing em linguagem C.

Avaliações

- Provas e testes
 - Não peça ajuda na prova, nenhum colega ou professor tem permissão de ajudar
 - Escritos
 - Sala de aula, sem consulta
 - Não é permitido o uso de computador, celular, iPod, etc. violações a esta regra serão punidas
 - No computador
 - Laboratório
 - Permitida a consulta à ajuda da ferramenta utilizada (sem acesso à Web ou mídias removíveis)

Critérios rígidos

- Fechou a porta, não entra mais (tolerância de 5 min)
- **É proibido o uso de celulares e smartphones durante as aulas**
- O critério de avaliação é claro e as avaliações são continuadas
 - Não adianta pedir nota, trabalho extra, etc..
- Notas NÃO serão arredondadas
 - O sistema SAGRES tem escala em décimos
- Provas substitutivas
 - Só com aprovação da Secretaria, mediante a apresentação de documentação pertinente
- Cola e uso de celulares/smartphones são punidos com rigor
 - Geralmente, suspensão imediata de uma semana

Material da disciplina

- <http://bit.ly/COM719>
 - Material de aula
 - Listas de exercício
 - Notas das avaliações

PIEF III – 2S2020

- Atividades do PIEF III **começam neste semestre**
 - **Contam nota** para o PIEF III do 2S2020
- **Integração do PIEF III da Produção e Computação**
 - Alunos da **Produção**
 - Levantarão **oportunidades** para sistemas na comunidade
 - Apresentarão **propostas** e modelagem do processo/negócio
 - **Professores** farão **análise** crítica de viabilidade
 - Alunos da **Computação** (grupos de 2 a 3 alunos)
 - **Assistirão** às apresentações
 - Apresentarão **relatório** contendo a análise crítica dos projetos
 - **Escolherão 3 projetos**, fazendo as devidas justificativas
 - Tais projetos **serão desenvolvidos no PIEF III do 2S2020**
 - Prestação de serviço/consultoria ou startup
 - Mas devem **iniciar** a busca por **aprendizado** de **tecnologias** já

Bibliografia

- **Bibliografia básica:**

ASCENCIO, Ana F. G.; ARAÚJO, Graziela S. Estruturas de Dados: algoritmos, análise da complexidade e implementações em Java e C/C++. São Paulo: Pearson Prentice Hall, 2010. 432 p.

EDELWEISS, Nina; GALANTE, Renata. Estruturas de Dados. Porto Alegre: Bookman, 2009. 262 p.

TENENBAUM, Aaron M.; LANGSAM, Yedidyah; AUGENSTEIN, Moshe J. Estruturas de Dados Usando C. São Paulo: Makron, 1995. 884 p.

- **Bibliografia complementar:**

GOODRICH, Michael T; TAMASSIA Roberto. Estruturas de dados e algoritmos em Java. Porto Alegre: Bookman, 2013. 735 p.

GUIMARÃES, Ângelo de Moura; LAGES, Newton Alberto de Castilho. Algoritmos e estruturas de dados. Rio de Janeiro: LTC, 1994. 216 p.

KERNIGHAN, Brian W.; RITCHIE, Dennis M. C, a linguagem de programação: padrão ANSI. Rio de Janeiro: Campus, 1989. 289 p.

SZWARCFITER, Jayme Luiz; MARKENSON, Lilian. Estruturas de dados e seus algoritmos, 3a ed. Rio de Janeiro: LTC, 2015. 346 p.

WIRTH, Niklaus. Algoritmos e Estruturas de Dados. São Paulo: LTC, 2012. 255p.

- **Materiais em formato eletrônico: tutoriais, man pages, etc.**

- **IMPORTANTE: vários livros estão disponíveis nas Bibliotecas Online**

Softwares de apoio

- GCC (GNU C Compiler) 32 ou 64 bits: gcc.gnu.org
 - Nativo no Linux: instalar pacote da distro
 - MinGW (Minimalist GNU for Windows): www.mingw.org
- Code::Blocks (IDE): www.codeblocks.org
 - No Linux: instalar pacote da distro
 - No Windows: procure a instalação (com MinGW, se preferir) em www.codeblocks.org/downloads
- Para casa
 - Baixar e instalar compilador + IDE