

1. Identificação

Curso:	ENGENHARIA DA COMPUTAÇÃO 02/2018		
Período letivo:	2º SEM/2019 UPT	Turno:	MULTIPERIÓDICO
Unidade curricular:	PESQUISA E ORDENAÇÃO		
Unidade:	CAMPUS AVANÇADO UBERABA PARQUE TECNOLÓGICO - UNIDADE I		
Tipo:	TEÓRICA/PRÁTICA		
Professor(es)	DANIELA RESENDE SILVA ORBOLATO	(PRINCIPAL)	
Carga horária:	80.00		
Duração aula:	40 minutos		

2. Ementa

Árvores múltiplas. Teoria de grafos. Algoritmos para pesquisa e ordenação em memória principal e secundária. Compressão de dados.

3. Objetivos

- Proporcionar aos alunos a capacidade de analisar e optar, frente a problemas relacionados à representação de informação, por estruturas de dados adequadas à sua solução, além da capacidade de desenvolver técnicas para a implementação de tais estruturas.
- Capacitar o aluno para solucionar problemas que envolvam pesquisa e ordenação em memória principal e secundária e compressão de dados.
- Capacitar o aluno para solucionar problemas que envolvam a organização e recuperação de informações armazenadas em arquivos externos por meio de algoritmos adequados de organização e busca.

4. Conteúdo Programático

Árvores Múltiplas – Árvores B, B*, B+

Grafos - Conceitos Básicos - Definição e notação de grafos. Subgrafos.

Grafos - Representação de Grafos - Matriz de adjacências. Listas.

Grafos - Caminhos e Circuitos - Caminhos, percursos, trajetos e ciclos de grafos. Caminhos e ciclos. Grafos conexos e desconexos. Distância entre vértices.

Grafos - Grafos valorados.

Algoritmos de busca em árvores/grafos

Busca em profundidade

Busca em largura

Busca gulosa

Menor caminho. Algoritmo de Dijkstra.

Métodos de Ordenação: inserção, seleção, borbulhamento, shell, quicksort

Algoritmos de busca em estruturas sequenciais

Busca sequencial

Busca com sentinela

Busca binária

Busca por interpolação

Compressão de dados: condições para a compressão de dados e a codificação de Huffman.

5. Metodologia

A metodologia de trabalho dará ênfase à compreensão do conteúdo aula a aula, que serão preparadas de modo que cada aula seja iniciada com um breve resumo da aula anterior e concluída com a identificação dos

tópicos principais vistos na aula.

Também com o objetivo de melhor adequar a apresentação às características dos alunos, pretende-se manter um diálogo aberto e saudável entre professora e alunos, para que as duas partes discutam suas impressões sobre o andamento da disciplina e, se houver necessidade, decidam mudanças sobre a forma de condução da disciplina.

A estratégia adotada será a de utilização de aulas expositivas interativas, iniciadas com um breve resumo da aula anterior, breve exposição do conteúdo e seguidas de momentos em que os alunos devem contribuir para construção do conhecimento, fazendo aplicações práticas dos conceitos apresentados implementados os algoritmos apresentados na linguagem C.

6. Recursos Didáticos

Para expor o conteúdo serão adotados slides, o quadro branco, data-show, os livros da bibliografia, páginas e tutoriais online e notas de aula. As atividades serão realizadas nos laboratórios de informática, dando ênfase ao aprender fazendo e procurando envolver as habilidades adquiridas em disciplinas anteriores do curso, como Algoritmos e Estruturas de dados, cujos conceitos precedem naturalmente a disciplina de Grafos.

Trabalhos intra e extraclasse e exercícios em sala serão passados aos alunos com o objetivo de fixar o aprendizado e promover a autonomia do aluno na busca pela solução dos problemas.

As listas de exercícios teóricas também constituem um recurso importante no aprendizado desta disciplina e terão por objetivo fazer com que o aluno retenha a base teórica-matemática do conteúdo, para depois transportá-la para implementação dos conceitos vistos.

As atividades serão realizados trabalhos de implementação em grupo de 2 a 3 alunos.

7. Avaliação (critérios, valores, procedimentos, recuperação)

Os alunos serão avaliados de forma contínua e integrada, considerando suas atuações no desenvolvimento de aplicações práticas na forma de trabalhos em grupo, seu empenho na apresentação e pontualidade nos trabalhos e suas notas nas provas.

A distribuição da pontuação se encontra especificada a seguir:

a) Avaliação escrita: 80 pontos

Deverão ser realizadas 3 avaliações ao longo do semestre, de acordo com o progresso do conteúdo, com uma divisão de pontuação de 25, 25 e 25 pontos.

Datas previstas:

setembro/2019

outubro/2019

dezembro/2019

b) Trabalhos de implementação: 25 pontos

É uma avaliação prática que compreende o desenvolvimento intra ou extraclasse, em grupos de algoritmos para representação de conceitos ou solução de problemas com os conceitos vistos. Os critérios de correção são correteza, completude, pontualidade e originalidade. Serão realizados no mínimo 2 trabalhos práticos e no máximo 4 trabalhos ao longo do semestre, com uma distribuição igualitária de pontos.

Com o objetivo de promover recuperação do aprendizado, todas as avaliações serão acompanhadas de vistas e discussões sobre o conteúdo das questões, para que o aluno possa observar seus erros, reavaliar e reformular os seus conceitos, sanando dificuldades e dúvidas.

Também neste item está a distribuição de pontos para Mostra de Produtos Tecnológicos, no valor de 10 pontos.

d) São previstas ainda provas de recuperação para as avaliações escritas. Os critérios para participação nessa recuperação são definidos pelas normas institucionais e regulamentação do colegiado, que atualmente define que os alunos com média inferior a 60% na avaliação poderão realizar uma recuperação da mesma e recebem a melhor nota entre a

prova original e sua recuperação.

8. Referências

BÁSICA

TENENBAUM, A.; LANGSAM, Y; AUGENSTEIN, M. Estrutura de dados usando C. 1 ed. Makron Books. São Paulo, 1995.

DROZDEK, A. Estrutura de dados e algoritmos em C++. 1 ed. São Paulo: Cengage Learning. 2002.

CORMEN, T. H. et al. Algoritmos: teoria e prática. Campus, 3 ed. Rio de Janeiro: Elsevier, 2012.

COMPLEMENTAR:

ZIVIANI, N. Projetos de Algoritmos: com implementações em Pascal e C. 3 ed. São Paulo: Cengage Learning, 2010.

KNUTH, D. E. The art of computer programming: sorting and searching. 2 ed. Addison Wesley, 1998. v. 3.

LIVADAS, P. E. File structures: 1 ed. Theory and Practice. Prentice Hall, 1990.

VELOSO, P. Estrutura de dados. 1 ed. Rio de Janeiro: Campus, 1999.

GOODRICH, M.T. et al. Projeto de algoritmos: fundamentos, análise e exemplos da internet. 1 ed. Porto Alegre: Bookman, 2004.

Cronograma das aulas

SEMANA	Nº AULAS PREVISTAS	DESCRIÇÃO	C. H. TEÓRICA	C.H. PRÁTICA
1	3	Apresentação da disciplina Apresentação da professora Revisão de árvores AVL	02:00	
2	3	Árvore B conceitos iniciais construção intuitiva algoritmo em pseudocódigo estrutura de classes da BTree	02:00	
3	3	Árvores B - algoritmo de inserção Construção da classe em JAVa	02:00	
4	3	Construção das classes-base da BTree e dos métodos buscaFolha e insereNaFolha para compor o método de inserção na BTree		02:00
5	3	Implementação do método de divisão na Btree Implementação do método de ajuste na divisão da raiz da Btree Implementação do método de inserção na Btree chamando os métodos anteriores.		02:00
6	3	Correções na inserção em BTree		02:00
7	3	Método para mostrar valores em ordem na BTree (recursivo)		02:00
8	3	Construção de árvores B+	02:00	
9	3	Árvore B+		02:00
10	3	Prova 1	02:00	
11	3	Árvore B*	00:40	01:20
12	3	Grafos - conceitos iniciais - representação em matriz de adjacências	02:00	

PLANO DE ENSINO

SEMANA	Nº AULAS PREVISTAS	DESCRIÇÃO	C. H. TEÓRICA	C.H. PRÁTICA
13	3	Implementação de Grafos com matriz de adjacências Implementação de grafos com lista de adjacências		02:00
14	3	Implementação de Grafos com matriz de adjacências Implementação de grafos com lista de adjacências		02:00
16	3	Busca em profundidade e em largura	00:40	01:20
17	3	Algoritmo de Dijkstra		02:00
18	3	Exercícios de grafos no URI		02:00
19	3	Hashing Conceitos Exemplo em Java: HashMap Exemplo de implementação de Hashing com colisões Exercícios de grafos no URI	02:00	
20	3	Prova 2	02:00	
21	3	Provas de recuperação	02:00	
20	60	Totais em horas	19,33	20,67

DANIELA RESENDE SILVA ORBOLATO

PRINCIPAL

JEFFERSON BEETHOVEN MARTINS

COORDENADOR DE CURSO