

Oferta de turma especial de GCC216 - Estrutura de Dados durante o período de quarentena

Professor responsável: Joaquim Quinteiro Uchôa

Introdução

A disciplina GCC216 - Estrutura de Dados é mantida pelo setor de Fundamentos de Programação do DCC e é oferecida para os cursos de Ciência da Computação e Sistema de Informação da UFLA. A disciplina busca cobrir os aspectos básicos de programação de computadores, com a seguinte ementa:

Tipos Abstratos de Dados (Listas, Pilhas e Filas); Técnicas de uso de Classes e Objetos; Árvores Binárias (AVL, Rubro-Negras, Heaps); Tabelas Hash; Busca e Ordenação Externa (Árvores B, B+ e B*, Mergesort externo).

Por envolver uma forma de raciocínio própria, o pensamento algorítmico, um tipo de resolução de problemas lógico-matemático, muitos alunos enfrentam dificuldades em seu aprendizado, especialmente no primeiro contato com a mesma. Cabe ressaltar nesse quesito, inclusive, que a disciplina é atualmente ofertada para alunos de segundo período nos respectivos cursos.

A resolução CEPE 473/18 define a possibilidade de ofertas de Turma-E semipresencial utilizando estratégias de mediação baseadas em tecnologias digitais de informação e comunicação, de acordo com seu capítulo número 128. De acordo com a instrução normativa da PRG número 13 de 2019, artigo segundo, cabe ao departamento a apresentação de proposta para oferta nos semestres letivos. Assim, em atendimento à demanda levantada pela Portaria 121/2020 - PRG, solicito a oferta de Turma-E COVID para a referida disciplina.

Objetivos

Possibilitar que GCC216 seja ofertada por meio de Turma-E durante a quarentena motivada pela epidemia de COVID-19, possibilitando que alunos reprovados possam continuar seus estudos e concluir a disciplina durante o período de isolamento social.

Metodologia

A disciplina será ofertada em uma única turma, com duração prevista para 6 de abril a 15 de maio, sendo que esse prazo ser prorrogado, uma vez que a intenção é que dure o período de quarentena. A matrícula deverá ser realizada até o dia 8 de abril, para que os alunos estejam no ambiente virtual até o dia 9 de abril. Ressaltamos que a oferta é com carga horária concentrada no período.

O professor responsável por esta solicitação será responsável pela produção e disponibilização de material didático (notas de aula, vídeos, etc.) e gerenciamento do

ambiente virtual de aprendizagem, bem como pelo acompanhamento e avaliação do aluno na disciplina, auxiliado pelo monitor da disciplina.

Além da bibliografia disponível de maneira eletrônica na Biblioteca da UFLA ou outros meios, os alunos terão a seu dispor diversos materiais disponibilizados pelo professor responsável, como slides e videoaulas, como já ocorre quando da preparação para avaliação suplementar (vide: <https://sites.google.com/ufla.br/roteiro-ed-suplementar/>).

Para interação com os alunos, será utilizada uma sala no Google Classroom da UFLA, ambiente que já é utilizado oficialmente, e com apoio da DGTI, nas turmas ofertadas em 2020/1. A sala será criada pelo professor responsável e os alunos matriculados serão inseridos na sala virtual pela DIREC ou pelo professor.

Os alunos serão avaliados principalmente por meio da ferramenta Dredd (<https://dredd.dcc.ufla.br/>), como já ocorre com as turmas presenciais. O Dredd é utilizado como uma ferramenta de apoio ao ensino de programação, possibilitando uma metodologia ativa e autônoma de aprendizagem. Além disso, ele também é utilizado para aplicação de provas, listas de exercícios tradicionais e listas de exercícios avaliativas.

Além da avaliação presencial, com percentual de 70%, os alunos serão avaliados por meio de listas de exercício (20%) e um projeto prático (10%) envolvendo conceitos da disciplina. Com exceção da avaliação presencial, todas as atividades podem ser realizadas remotamente e no local de estadia do aluno, desde que esse possua acesso adequado à internet.

Estruturas de Dados - GCC216

Plano de Curso - Turma E - Quarentena

Objetivos

Ao término da disciplina, o aluno deve ser capaz de:

- usar técnicas básicas de programação orientada a objetos;
- reconhecer a necessidade de estruturas de dados básicas em memória;
- entender o funcionamento, a funcionalidade e as principais aplicações das estruturas de dados básicas em memória, a saber: pilha, fila, lista, heap, tabela hash, árvores binárias;
- reconhecer a necessidade e quais as principais formas de balanceamento de árvores binárias;
- entender o funcionamento, a funcionalidade e as principais aplicações de estruturas de dados de busca balanceadas, como árvores AVL, árvores rubro-negras e árvores B;
- reconhecer a necessidade de algoritmos de ordenação e estruturas de dados otimizadas para uso em memória secundária;
- entender o funcionamento, a funcionalidade e as principais aplicações de métodos de ordenação em memória secundária;
- entender o funcionamento, a funcionalidade e as principais aplicações de estruturas de dados básicas de busca em memória secundária, a saber: árvores 2-3 e 2-3-4, árvores B, B+ e B*;
- ser capaz de solucionar problemas reais com o uso das estruturas mais adequadas ao problema em questão.

Cronograma de Atividades

Carta horária da disciplina: 68 aulas teóricas e 34 aulas práticas - 102 horas

Oferta com carga horária concentrada

Aulas	Período	Conteúdo
Semana 01 (14 horas)	06/04 a 12/04	Apresentação da disciplina, do professor, metodologia de ensino e forma de avaliação. Tipos abstratos de dados (TADs). Classes e objetos: conceitos básicos, campos e métodos; visibilidade/encapsulamento. Construtores e destrutores; alocação dinâmica de memória; construtor de cópia; sobrecarga de métodos. Classes e objetos: sobrecarga de operadores. Exemplos.
Semana 02 (14 horas)	13/04 à 19/04	Estrutura de dado pilha: visão geral e implementação. Estrutura de dado fila: visão geral e implementação. Conceitos de árvores e grafos. Estrutura de dado heap: conceitos básicos e estratégias de implementação. Estrutura de dado torneio.

Semana 03 (18 horas)	20/04 à 26/04	Estrutura de dados lista: visão geral (dinâmica/arranjo); operações em listas; estratégias de implementação, iteradores. Estruturas baseadas em listas: deque, vetores expansíveis e sequence sets. Tabelas hash: conceitos básicos e formas de implementação. Hash por encadeamento. Tabelas hash: formas alternativas de tratamento de colisão. Hash por endereçamento aberto.
Semana 04 (18 horas)	27/04 à 03/05	Árvores binárias de busca: conceitos básicos, estratégias de implementação. Estrutura de dado Trie. Árvores AVL: conceitos básicos, principais métodos e estratégias de implementação. Árvores rubro-negras: conceitos básicos, principais métodos e estratégias de implementação.
Semana 05 (18 horas)	04/05 a 10/05	Árvores 2-3 e 2-3-4: conceitos básicos e estratégias de implementação. Estruturas de dados e algoritmos em memória secundária: visão geral e problemas. Árvores B: conceitos básicos e principais métodos. Árvores B em memória: formas de implementação. Árvores B+ e B*: visão geral e formas de implementação.
Semana 06 (17 horas)	11/05 a 15/05	Ordenação externa: visão geral, mergesort em memória, mergesort externo. Intercalação multi-caminhos. Ordenação externa: substituição por seleção e intercalação polifásica. Ordenação externa: estratégias de implementação. Apresentação do projeto prático.
(3 horas)	Em aberto	Avaliação presencial a ser realizada na primeira segunda-feira após o retorno às aulas.

Metodologia de Ensino:

Disciplina possui ênfase em aplicação prática de estruturas de dados, mesmo em aulas teóricas. O computador é o laboratório preferencial para aplicação dos conteúdos, com implementações em C/C++, usando técnicas de programação orientada a objetos. O aprendizado é concretizado por meio de resolução de listas de exercícios diversos, utilizando a ferramenta Dredd (<https://dredd.dcc.ufla.br/>).

Avaliação:

- 1 avaliação presencial prática (utilizando o Dredd), valendo 70 pontos.
- Projeto Prático: 10 pontos.
- Listas de exercícios: 20 pontos divididos em quatro listas de exercícios. As três primeiras listas, valem 4 pontos cada. A última lista, envolvendo árvores B e ordenação externa, vale 8 pontos. Será utilizada ferramenta para identificação de similaridade e as listas em que forem identificadas cópias terão nota atribuída igual a 0 (zero) e as evidências serão enviadas à PRG para abertura de processo disciplinar.

Bibliografia

DEITEL, H. M. & DEITEL, P. J. **C++: como programar**. 5.ed. São Paulo, Pearson, 2006. *(Disponível como e-book na biblioteca da UFLA)*.

EDELWEISS, N. & LIVI, M. A. C. **Algoritmos e programação com exemplos em Pascal e C**. Porto Alegre: Bookman, 2014. *(Disponível como e-book na biblioteca da UFLA)*.

FARRER, H. et al. **Programação Estruturada de Computadores: Algoritmos Estruturados**. 3.ed. Rio de Janeiro: LTC, 1999.

FORBELLONE, A. L. V & EBERSPÄCHER, H. F. **Lógica de Programação: A Construção de Algoritmos e Estruturas de Dados**. 3.ed. São Paulo: Prentice-Hall, 2005. *(Disponível como e-book na biblioteca da UFLA)*.

PIVA JUNIOR, D.; NAKAMITI, G. S.; ENGELBRECHT, A. de M. & BIANCHI, F. **Algoritmos e Programação de Computadores**. Rio de Janeiro, Campus, 2012.

PUGA, S. & RISSETTI, G. **Lógica de programação e estruturas de dados: com aplicações em Java**. 3.ed. São Paulo: Pearson, 2016. *(Disponível como e-book na biblioteca da UFLA)*.