

Estruturas de Dados

A. G. Silva, A. von Wangenheim, J. E. Martina

01 de agosto de 2018

Organização da disciplina

- Objetivos
- Metodologia
- Avaliação
- Material didático
- Realização dos trabalhos
- Programa da disciplina
- Bibliografia
- Contato com os docentes

Objetivos da disciplina

- Aprender as técnicas básicas de organização de dados em um computador sob a forma de estruturas de dados:
 - Em memória;
 - Em meios externos.
- Aprender a manipular e implementar estruturas de dados utilizando diversas técnicas de programação atualmente em uso.
- Aprender a projetar e a avaliar a eficiência de algoritmos complexos.

Objetivos da disciplina

O aluno deverá:

- 1 Adquirir o conhecimento necessário para desenvolver programas de computador eficazes e eficientes para a solução de problemas complexos.
- 2 Adquirir a experiência prática de desenvolvimento de programas de computador para desenvolver e estimar adequadamente a dificuldade de desenvolvimento de programas complexos.

Importante!

O domínio de Estruturas de Dados será indispensável para praticamente tudo o que você for aprender daqui para frente.

- Disciplina eminentemente prática, com ênfase na aplicação dos conhecimentos;
- Aulas teóricas alternadas com aulas práticas;
 - Serão vistos aspectos de modelagem, algoritmos, técnicas de implementação e bases teóricas dos modelos.
- Aulas práticas em laboratório:
 - Ministradas pelo estagiário de docência e pelo professor;
 - Sempre iniciarão com uma complementação visando a implementação do que foi visto na aula teórica;
 - Objetivam prover oportunidade para o aluno fixar o que foi visto e aprimorar as suas técnicas de programação.

Metodologia de ensino – implementação

- Metodologias de implementação/codificação:
 - Serão vistas em paralelo às estruturas de dados;
 - Reveremos a evolução histórica das técnicas atualmente em uso;
- Técnicas de implementação de programas:
 - Programação Orientada a Objetos;
 - Meta-programação;
 - Uso de recursos avançados da linguagem;
- Linguagens de programação adequadas às técnicas:
 - Discussão da adequação das linguagens;
 - Exemplos em outras linguagens.

Critérios de avaliação

- 7 Notas (Pesos: 2,1,1,1,2,1,2):
 - 2: média dos pequenos trabalhos (entre 12 e 18);
 - 1: projeto de implementação I;
 - 1: teste teórico I;
 - 1: teste prático de implementação I;
 - 2: projeto de implementação II;
 - 1: teste prático de implementação II;
 - 2: teste teórico II.

Avaliação das implementações:

Automatizada pelo Moodle segundo critérios definidos no enunciado.

Recuperação

- Conceito “Menção I” somente será cogitado de acordo com a legislação da UFSC;
- A recuperação pode não ocorrer porque a disciplina é mais que 50% prática;
- A recuperação será realizada através de um trabalho de implementação seguido de entrevista;
 - O trabalho de recuperação será individual;
 - O trabalho de recuperação terá peso igual ao da média do semestre, de acordo com a legislação da UFSC.

- Material da disciplina disponível na Internet (<http://moodle.ufsc.br/>):
 - Fio condutor das aulas para o aluno;
 - Não usar como única fonte;
 - Cronogramas, listas de trabalhos e datas;
 - Plano de ensino;
 - Transparências de aula;
 - Arquivos de dados para exercícios e programas-exemplo;
 - Links para sites com manuais e documentação útil para programação.
- Livros indicados no plano de ensino.

Realização dos trabalhos

- Os trabalhos pequenos são todos individuais;
- Equipes de 2 pessoas (somente trabalhos grandes):
 - É importante aprender a realizar trabalhos de implementação em equipe;
- Implementação:
 - ANSI C++ - OOP;
 - Plataforma homologada é somente o VPL;
- Ferramentas & máquinas:
 - Aproveite as aulas de laboratório;
 - Linux para C/C++;
 - Compiladores: gcc / g++

Entrega de trabalhos

- Data:
 - Deverão ser entregues na data especificada no Moodle;
 - São aceitos com atraso, mas com 50% de desconto na nota;
 - Projetos de implementação também;
- Serão aceitos os trabalhos que:
 - Satisfizerem as especificações / enunciados dados;
 - Utilizarem a linguagem e a técnica de programação especificadas;
 - Utilizarem codificação de acordo com os padrões internacionais para a linguagem especificada;
 - Estiverem devidamente documentados.
 - Executarem na plataforma homologada.

Controle de frequência

- O controle de frequência segue as regulamentações da UFSC para cursos presenciais;
- É obrigatório 75% de presença para aprovação;
- O controle de frequência vai ser feito diretamente no Moodle da disciplina;

Programa da disciplina

- Introdução e Aspectos Gerais da Disciplina;
- Introdução à Programação C++;
- Listas, Pilhas e Filas utilizando Vetores (arrays);
- Gerência e Alocação Dinâmica de Memória;
- Lista Encadeada;
- Fila e Pilha Baseados em Listas Encadeadas;
- Lista Duplamente Encadeada;
- Lista Circular;

Programa da disciplina

- Conceitos de Complexidade de Algoritmos;
- Introdução a Árvores;
- Árvores I;
- Árvores de Busca II;
- Hashing;
- Gerência de Arquivos;
- Ordenação.

1 Material de Estruturas de Dados:

- Aguilar: Programação em C++: algoritmos, estruturas de dados e objetos. McGraw Hill;
- Horowitz: Estruturas de Dados. Ed. Campus;
- Wirth: Data Structures + Algorithms = Programs;
- Outros livros, a maioria em inglês, todos disponíveis na BU ou no INE;

2 Material sobre C++:

- Livros sobre Estruturas de Dados em “C” e C++;
- Material sobre linguagem C++;

3 Bibliografia de ferramentas de programação:

- Livros sobre Unix / Linux;
- Manuais de Editores;
- Manuais de Programação;
- Material Online.

Contato e marcação de horários

- Monitoria: será disponibilizado um horário por semana para atendimento pelo estagiário de docência ou bolsista da disciplina;
- Procedimento:
 - Envie e-mail explicando seu problema;
 - Utilize sempre o “Fórum de discussão”;
 - Se possível, o problema será resolvido por e-mail;
 - Caso contrário, será indicado o horário de atendimento.
- Marcação de horários
 - Professores por mensagem no moodle;
 - Estagiário de docência e bolsista por e-mail divulgado no Moodle.

Perguntas????





Este trabalho está licenciado sob uma Licença Creative Commons Atribuição 4.0 Internacional. Para ver uma cópia desta licença, visite

<http://creativecommons.org/licenses/by/4.0/>.

