



PLANO DE APRENDIZAGEM

1. DADOS DE IDENTIFICAÇÃO			
Curso: Bacharelado em Sistemas de Informação			
Disciplina: Algoritmos e Estruturas de Dados		Código: SIF33	
Professor: Mirthys Marinho do Carmo Melo		e-mail: mirthys.melo@fasete.edu.br	
CH Teórica: 80	Prática: 20	CH Total: 100	Créditos: 05
Pré-requisito(s): Linguagem de Programação I			
Período: III		Ano: 2019.2	

2. EMENTA:

Conceitos avançados de programação: recursão, ponteiros, alocação dinâmica, tipos abstratos de dados. Complexidade de Algoritmos: conceitos básicos. Listas Lineares sequenciais e encadeadas. Listas simplesmente encadeadas, listas duplamente encadeadas, listas circulares, multilistas. Listas lineares restritas: pilhas, filas. Árvores. Árvores binárias, árvores binárias de busca, árvores balanceadas tipo AVL. Classificação interna de dados: classificação por seleção, método da bolha, classificação por inserção, classificação por Quick Sort.

3. COMPETÊNCIAS E HABILIDADES DA DISCIPLINA:

Modelar e implementar soluções de Tecnologia de Informação em variados domínios de aplicação, através do desenvolvimento de soluções para problemas apresentados durante as aulas;

Gerenciar equipes de trabalho no desenvolvimento e evolução de Sistemas de Informação, utilizando a dinâmica de criação de grupos de trabalhos para o desenvolvimento de soluções computacionais;

Aplicar conceitos, métodos, técnicas e ferramentas de gerenciamento de projetos em sua área de atuação, com a apresentação de projetos que utilizem as estruturas de dados estudadas durante a disciplina;

Aprimorar experiência das partes interessadas na interação com a organização, incluindo aspectos da relação humano-computador através do desenvolvimento de programas que tenha como propósito facilitar as tarefas dos usuários;

Especificar, projetar e implementar software para sistemas de informação, utilizando as estruturas de dados estudadas durante a disciplina;

Avaliar a qualidade de processos e produtos de software para Sistemas de Informação, através de testes de mesa e de correção dos programas implementados na disciplina;

Identificar oportunidades de mudanças e projetar soluções usando tecnologias da informação nas organizações, através de propostas de soluções possíveis para problemas reais;



4. OBJETIVO GERAL DA APRENDIZAGEM:

Desenvolver a competência de modelar soluções computacionais de forma mais organizada e estruturada, habilitando o aluno a implementar códigos padronizados e mais robustos .

5.1 -PRIMEIRA ETAPA

5.1.1 – CONTEÚDOS PRESENCIAIS (40 Horas Aulas)

- Revisitar dos tipos básicos de uma linguagem de programação (5h);
- Revisitar dos conceitos de vetores e matrizes (5h);
- Fundamentar os conceitos avançados de programação: recursão, ponteiros, alocação dinâmica (10h);
- Apresentar o conceito de tipos abstratos de dados (10h);
- Compreender o conceito de Listas Lineares sequenciais e encadeadas: listas simplesmente encadeadas, listas duplamente encadeadas e listas circulares (10h).

5.1.2 – CONTEÚDOS NO AMBIENTE VIRTUAL DE APRENDIZAGEM (10 Horas Aulas)

- Desafios de conhecimento com assuntos correlatos aos conteúdos iniciais da disciplina (3h);
- Disponibilização de lista de exercícios práticos de programação (4h);
Disponibilização de material para os alunos desenvolverem uma aula expositiva discursiva para ser ministradas presencialmente (3h).

5.2 -SEGUNDA ETAPA

5.2.1 – CONTEÚDOS PRESENCIAIS (40 Horas Aulas)

- Entender o conceito de listas lineares restritas: pilhas e filas (10h);
- Demonstrar, na prática, a utilização de estruturas de dados abstratas utilizando a linguagem C (5h);
- Compreender o conceito de árvores (10h);
- Entender a aplicabilidade de árvores em problemas computacionais (5h);
- Apresentar exemplos de implementação de Árvores Binárias (5h);
- Apresentar os paradigmas de classificação Interna de Dados (5h).

5.2.2 – CONTEÚDOS NO AMBIENTE VIRTUAL DE APRENDIZAGEM (10 Horas Aulas)

- Disponibilização de lista de exercícios práticos de programação (4h);
- Desenvolvimento do projeto da disciplina (6h).

6.: METODOLOGIA DO TRABALHO:

6.1-1º ETAPA

6.1.1 – Metodologias Ativas Presenciais



- Aulas expositivas discursivas;
- Gameificação, com o uso do socrative.

6.1.2- Metodologias baseadas nas Tecnologias

- **WebQuest:** atividades direcionadas aos conteúdos iniciais da disciplina;
- **Aprendizagem Baseada em Problemas:** lista de exercícios práticos de programação;
- **Sala de Aula Invertida:** disponibilização de material para os alunos desenvolverem uma aula expositiva discursiva para ser ministradas presencialmente.

6.2-1º ETAPA

6.2.1 – Metodologias Ativas Presenciais

- Aulas expositivas discursivas;
- Correção interativa dos exercícios;

6.2.2- Metodologias baseadas nas Tecnologias

- **Aprendizagem Baseada em Problemas:** lista de exercícios práticos de programação;
- **Aprendizagem Orientada a Projetos:** desenvolvimento de um projeto desktop, utilizando a plataforma Dev C++ e a linguagem C, englobando todo o assunto da disciplina.

7. RECURSOS:

<input checked="" type="checkbox"/> Sala (comum)	<input type="checkbox"/> Sala Configuração Flexível	<input checked="" type="checkbox"/> Laboratório(s) - agendar
<input checked="" type="checkbox"/> Práticas em Campo	<input checked="" type="checkbox"/> Kit multimídia	<input type="checkbox"/> Outros (informar)

8. SISTEMA DE AVALIAÇÃO:

AVALIAÇÃO:

- **1ª Etapa:**

Avaliação Processual: lista de exercícios incrementais sobre o assunto dado (10,0);

Avaliação Institucional (Modelo ENADE) (10,0);

Avaliação da Aprendizagem Baseada nas Tecnologias: WebQuest (3,0), sala de aula invertida (3,0) e aprendizagem baseada em problemas (4,0).

- **2ª Etapa:**

Avaliação Processual: lista de exercícios incrementais sobre o assunto dado (10,0);



Avaliação Institucional (Modelo ENADE) (10,0);

Avaliação da Aprendizagem Baseada nas Tecnologias: aprendizagem baseada em problemas (5,0) e aprendizagem orientada a projetos (5,0).

09. ATENDIMENTO EXTRA CLASSE:

Diariamente, através do endereço eletrônico: mirthys.melo@fasete.edu.br.

Semanalmente, mediante pré-agendamento.

10. BIBLIOGRAFIA BÁSICA:

CORMEN, Thomas H et al. Algoritmos: teoria e pratica. Rio de Janeiro: Campus 2002. SCHILDT, Herbet; MAYER, Roberto Carlos. C completo e total. São Paulo: Makron Books, 1997. VELOSO, Paulo; TOSCANI, Laira Vieira. Complexidade de algoritmos. 2ª ed. Porto Alegre: Sagra-Luzzatto, 2005.

11. BIBLIOGRAFIA COMPLEMENTAR:

AZEVEDO, Paulo Alberto de. Tabelas: organização e pesquisa. Porto Alegre: Sagra-Luzzatto, 2003. FORD, William. TOPP, William. Data Structures with C++ using STL. EUA: Prentice-Hall, 2002. FOROUZAN, Behrouz A.; GILBERT, Richard F. Data Structures: a pseudocode approach with C++. Brooks Cole, 2001. JAMSA, Kris; KLAMBER, Lars; SANTOS, Jeremias René D. Pereira dos. Programando em C/C++: a bíblia. São Paulo: Pearson Education, 1999. K WIRTH, Niklaus, Algoritmos e Estruturas de Dados. Rio de Janeiro: LTC, 1999.

12. LEITURA COMPLEMENTAR:

AHO, Alfred V.; HOPCROFT, John. E.; ULLMAN, Jeffrey. D. Data Structures and algorithms. USA: Addison Wesley Publishing Company, 1987. AZEVEDO, Paulo Alberto de. Tabelas: organização e pesquisa. Porto Alegre: Sagra-Luzzatto, 2003. FOLK, Michael J. File Structures: an object-oriented approach with C++. EUA: Addison Wesley Publishing Company, 1998.

13. INFORMAÇÕES COMPLEMENTARES:

Em complementação de carga horária, serão acrescentadas 20 horas, correspondente a 20% da carga horária da disciplina, referente ao acompanhamento das atividades descritas na Metodologia do Trabalho, através da ferramenta de gerência de projeto disponível do *Microsoft Teams*.

14. APROVAÇÃO:



FASETE
FACULDADE SETE DE SETEMBRO
PAULO AFONSO - BA

ORGANIZAÇÃO SETE DE SETEMBRO DE CULTURA E ENSINO LTDA
Recredenciada pela Portaria / MEC n.º 881/2016 - D.O.U. 15/08/2016
CNPJ: 03.866.544/0001-29 e Inscrição Municipal n.º 005.312-3

Aprovado em ____/____/____

Homologado em ____/____/____

COORDENADOR(A)

GERÊNCIA ACADÊMICA

OBS: As datas das avaliações poderão sofrer alterações de acordo com o disciplinado pela secretaria acadêmica da FASETE.