



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL**

## **PLANO DE ENSINO**

### **1. IDENTIFICAÇÃO**

**Curso: Ciência da Computação**

**Componente curricular: GEX015 - Estrutura de Dados I**

**Fase: Segunda (Noturno)**

**Ano/semestre: 2018/1**

**Número da turma: 21319**

**Número de créditos: 4**

**Carga horária – Hora aula: 72**

**Carga horária – Hora relógio: 60**

**Professor: Daniel Di Domenico**

**Atendimento ao Aluno: Segundas-feiras (16h até 18h e 30min)**

**Sala 220 – Bloco dos Professores**

### **2. OBJETIVO GERAL DO CURSO**

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional.

### **3. EMENTA**

Alocação dinâmica de memória. Variáveis estáticas e dinâmicas. Listas lineares e suas generalizações: listas ordenadas, listas encadeadas, pilhas e filas. Complexidade de algoritmos. Algoritmos de busca e ordenação. Implementações com linguagem imperativa estruturada.

## 4. OBJETIVOS

### 4.1 GERAL

Utilizar estruturas de dados básicas e avançadas para a solução de problemas computacionais e analisar as soluções propostas para determinar suas complexidades.

### 4.2 ESPECÍFICOS

- Identificar a melhor estrutura de dados para resolver um determinado problema.
- Identificar se um algoritmo é eficiente.
- Aplicar as melhores estratégias de busca e ordenação conforme o contexto.

## 5. CRONOGRAMA E CONTEÚDOS PROGRAMÁTICOS

Total Parc.	Encontro	Conteúdo
4	1 e 2	Introdução à disciplina. Ponteiros.
8	3 e 4	Ponteiros.
12	5 e 6	Alocação Dinâmica.
16	7 e 8	Pilhas, filas, listas encadeadas, listas duplamente encadeadas, listas circulares.
20	9 e 10	Pilhas, filas, listas encadeadas, listas duplamente encadeadas, listas circulares.
24	11 e 12	Pilhas, filas, listas encadeadas, listas duplamente encadeadas, listas circulares.
28	13 e 14	Pilhas, filas, listas encadeadas, listas duplamente encadeadas, listas circulares.
32	15 e 16	Pilhas, filas, listas encadeadas, listas duplamente encadeadas, listas circulares.
36	17 e 18	<b>Primeira Avaliação Escrita P1.</b> Feedback P1.
40	19 e 20	<b>Recuperação da P1 (PR1).</b> Conceitos de complexidade de algoritmos.
44	21 e 22	Conceitos de complexidade de algoritmos. Métodos de ordenação de complexidade quadrática (Bubble Sort, Insertion Sort, Selection Sort).
48	23 e 24	Métodos de ordenação de complexidade quadrática (Bubble Sort, Insertion Sort, Selection Sort).
52	25 e 26	Métodos de ordenação de complexidade logarítmica (Quick Sort, Merge Sort).

<b>Total Parc.</b>	<b>Encontro</b>	<b>Conteúdo</b>
56	27 e 28	Métodos de ordenação de complexidade logarítmica (Quick Sort, Merge Sort).
60	29 e 30	Métodos de busca de complexidade linear e logarítmica (sequencial, binária).
64	31 e 32	Métodos de busca de complexidade constante (hashing).
68	33 e 34	<b>Segunda Avaliação Escrita P2.</b> Feedback P2.
72	35 e 36	<b>Recuperação da P2 (PR2).</b> Curiosidades e aplicações de Estrutura de Dados.

## 6. PROCEDIMENTOS METODOLÓGICOS

Conduzir a disciplina com aulas expositivas e debatidas, evoluindo em tópicos específicos para exercícios práticos e contextualizando com base em publicações atualizadas. Atividades em laboratório visando exercitar os conteúdos explorados.

## 7. AVALIAÇÃO DO PROCESSO ENSINO-APRENDIZAGEM

Para esta etapa serão utilizadas de abordagens tais como: avaliações teóricas e práticas, exercícios extra-classe e trabalhos de implementação. O resultado é expresso sob a forma de nota que varia de 0 (zero) a 10 (dez), com intervalos de 0,1 (um décimo). A média final mínima para aprovação é 6,0 (seis). A frequência mínima para aprovação é de 75,0% (setenta e cinco por cento).

A Média Final (MF) será composta por avaliações escritas (Ps) e trabalhos (TRs) sendo calculado da seguinte forma:

$$MF = P1 * 0,3 + P2 * 0,3 + TR * 0,4$$

sendo TR calculado da seguinte forma:

$$TR = (TR1 + TR2 + \dots + TRn) / n$$

onde TRn represente a nota de um trabalho, variando de zero (0) a dez (10).

Observações das avaliações:

- Os trabalhos (TRs) e as avaliações escritas (Ps) serão individuais;
- As avaliações escritas (Ps) serão sem consulta;
- Caso seja detectado plágio o aluno recebe zero no trabalho em questão, além disso, por demonstrar prática não aceitável o caso será levado ao conhecimento do colegiado;
- As avaliações corrigidas serão entregues aos alunos e os resultados serão analisados e

discutidos de forma coletiva.

- O formato dos trabalhos será definido pelo professor no decorrer do processo de ensino-aprendizagem, tendo em vista o caráter processual da avaliação.
- Em relação à avaliação dos trabalhos, os seguintes elementos serão levados em consideração:
  - Funcionamento correto (o programa precisa cumprir seu objetivo conforme a descrição do trabalho);
  - Legibilidade do código (nomes de classes com a primeira letra maiúscula, métodos e propriedades, identificação correta, etc);
  - Comentários (o código fonte deve conter um bloco de comentário no começo informando o propósito do programa e o nome/email do seu autor).
  - Haverá um desconto de 50% da nota do trabalho no caso de entrega em atraso, com prazo máximo de três dias;
  - Programas que não compilarem receberão nota zero instantânea (nenhuma avaliação será realizada).

### **7.1 RECUPERAÇÃO: NOVAS OPORTUNIDADES DE APRENDIZAGEM E AVALIAÇÃO**

Para cada Ps será ofertada uma prova de recuperação (RPs). Todos os estudantes têm direito a fazer as provas de recuperação, porém haverá reposição apenas da nota da prova, não havendo reposição das notas dos trabalhos.

As provas de recuperação seguirão alguns critérios que serão pontuados a seguir:

- Se a nota da prova de recuperação RP for maior do que a nota da prova P, ocorre substituição da nota.
- Se a nota da prova de recuperação RP for menor do que a nota da prova P, é feita média aritmética simples entre as notas de P e RP.
- O estudante terá 10 minutos após o início da RP para optar em fazê-la ou não, de modo que se optar em não fazê-la automaticamente permanecerá com a nota da prova P.

## **8. REFERÊNCIAS**

### **8.1 BÁSICA**

EDELWEISS, N.; GALANTE, R. Estruturas de Dados. Porto Alegre: Bookman, 2009.

CORMEN, T.; LEISERSON, C.; RIVEST, R.; STEIN, C. Algoritmos: Teoria e Prática. Rio de Janeiro: Campus, 2002.

ZIVIANI, N. Projeto de Algoritmos com Implementações Pascal e C. 2. ed. São Paulo: Thomson Pioneira, 2004.

## **8.2    *COMPLEMENTAR***

CELES, W.; CERQUEIRA, R.; RANGEL, J. L. Introdução a Estrutura de Dados: uma Introdução com Técnicas de Programação em C. Rio de Janeiro: Campus, 2004.

WIRTH, N. Algoritmos e Estruturas de Dados. Rio de Janeiro: LTC, 1989.

OLIVEIRA CRUZ, A. J. de. Curso de linguagem C, apostila, Rio de Janeiro, Março 2011.

---

Profº Daniel Di Domenico

---

Coordenador do curso

Emílio Wuerges

SIAPE: 2052314